

MySQL Installation and Setup on Ubuntu 22.04

1. To begin, download the .deb apt repository file:

```
mysqlasrvadmin@mySQLsrv:~$ wget https://dev.mysql.com/get/mysql-apt-config_0.8.24-1_all.deb
--2022-11-12 14:49:17-- https://dev.mysql.com/get/mysql-apt-config_0.8.24-1_all.deb
Resolving dev.mysql.com (dev.mysql.com)... 104.70.53.122, 2600:1408:5400:4b3::2e31, 2600:1408:5400:4a7::2e31
Connecting to dev.mysql.com (dev.mysql.com)|104.70.53.122|:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: https://repo.mysql.com//mysql-apt-config_0.8.24-1_all.deb [following]
--2022-11-12 14:49:17-- https://repo.mysql.com//mysql-apt-config_0.8.24-1_all.deb
Resolving repo.mysql.com (repo.mysql.com)... 23.34.248.44
Connecting to repo.mysql.com (repo.mysql.com)|23.34.248.44|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 18048 (18K) [application/x-debian-package]
Saving to: 'mysql-apt-config_0.8.24-1_all.deb'

mysql-apt-config_0.8.24-1_all.deb      100%[=====]
2022-11-12 14:49:17 (239 MB/s) - 'mysql-apt-config_0.8.24-1_all.deb' saved [18048/18048]

mysqlasrvadmin@mySQLsrv:~$ ls
mysql-apt-config_0.8.24-1_all.deb
```

2. Next, install the repositories for MySQL:

```
mysqlasrvadmin@mySQLsrv:~$ sudo dpkg -i mysql-apt-config_0.8.24-1_all.deb
Selecting previously unselected package mysql-apt-config.
(Reading database ... 60622 files and directories currently installed.)
Preparing to unpack mysql-apt-config_0.8.24-1_all.deb ...
Unpacking mysql-apt-config (0.8.24-1) ...
Setting up mysql-apt-config (0.8.24-1) ...
Warning: apt-key should not be used in scripts (called from postinst maintainer script of the package mysql-apt-config)
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
OK
mysqlasrvadmin@mySQLsrv:~$ sudo apt-get update
Hit:1 http://azure.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://azure.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Get:3 http://azure.archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]
Get:4 http://azure.archive.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://azure.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [698 kB]
Get:6 http://azure.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [417 kB]
Get:7 http://repo.mysql.com/apt/ubuntu jammy InRelease [15.2 kB]
```

3. Install MySQL:

```
mysqlasrvadmin@mySQLsrv:~$ sudo apt-get install mysql-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libmecab2 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client mysql-common mysql-community-client mysql-community-client-core
  mysql-community-client-plugins mysql-community-server mysql-community-server-core
The following NEW packages will be installed:
  libmecab2 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client mysql-common mysql-community-client mysql-community-client-core
  mysql-community-client-plugins mysql-community-server mysql-community-server-core mysql-server
0 upgraded, 12 newly installed, 0 to remove and 0 not upgraded.
Need to get 37.9 MB of archives.
After this operation, 272 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://azure.archive.ubuntu.com/ubuntu jammy/main amd64 libmecab2 amd64 0.996-14build9 [199 kB]
Get:2 http://azure.archive.ubuntu.com/ubuntu jammy/main amd64 mecab-utils amd64 0.996-14build9 [4850 B]
```

4. Confirm installation by checking the status of the MySQL service:

```

Running kernel seems to be up-to-date.

Restarting services...
Service restarts being deferred:
systemctl restart networkd-dispatcher.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
mysqlsradmin@mysqlsrv:~$ systemctl status mysql.service
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2022-11-12 14:54:03 UTC; 27s ago
     Docs: man:mysqld(8)
           http://dev.mysql.com/doc/refman/en/using-systemd.html
   Main PID: 7091 (mysqld)
    Status: "Server is operational"
     Tasks: 39 (limit: 4095)
    Memory: 368.2M
       CPU: 1.138s
    CGroup: /system.slice/mysql.service
           └─7091 /usr/sbin/mysqld

Nov 12 14:54:01 mysqlsrv systemd[1]: Starting MySQL Community Server...
Nov 12 14:54:03 mysqlsrv systemd[1]: Started MySQL Community Server.

```

Successful installation of MySQL^

5. Run `mysql_secure_installation` to configure better initial security. Set a strong password and validate strength of user passwords, change root password, disable anonymous users, restrict root login to localhost, disable the test database, and reload privilege tables:

```

mysqlsradmin@mysqlsrv:~$ mysql_secure_installation

Securing the MySQL server deployment.

Enter password for user root:

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: Y

There are three levels of password validation policy:

LOW      Length ≥ 8
MEDIUM  Length ≥ 8, numeric, mixed case, and special characters
STRONG Length ≥ 8, numeric, mixed case, special characters and dictionary
        file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 2
Using existing password for root.

Estimated strength of the password: 0
Change the password for root ? ((Press y|Y for Yes, any other key for No) : Y

New password:

Re-enter new password:

Estimated strength of the password: 100

```

```

Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No) : y
By default, a MySQL installation has an anonymous user, allowing anyone to log into MySQL without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : y
Success.

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y
Success.

By default, MySQL comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) : y
- Dropping test database...
Success.
- Removing privileges on test database...
Success.

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

```

```

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

```

All done!

Successful configuration of the MySQL Secure installation tool^

6. Check LAN functionality using ICMP:

```

lampyadmin@LAMPy:~$ ping -c4 10.0.1.4
PING 10.0.1.4 (10.0.1.4) 56(84) bytes of data.
64 bytes from 10.0.1.4: icmp_seq=1 ttl=64 time=1.05 ms
64 bytes from 10.0.1.4: icmp_seq=2 ttl=64 time=1.03 ms
64 bytes from 10.0.1.4: icmp_seq=3 ttl=64 time=1.06 ms
64 bytes from 10.0.1.4: icmp_seq=4 ttl=64 time=1.20 ms

— 10.0.1.4 ping statistics —
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.032/1.084/1.201/0.068 ms
lampyadmin@LAMPy:~$

```

```

mysqlasrvadmin@MySQLsrv:~$ ping -c4 10.0.0.4
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=64 time=1.00 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=64 time=1.00 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=64 time=1.12 ms
64 bytes from 10.0.0.4: icmp_seq=4 ttl=64 time=0.973 ms

— 10.0.0.4 ping statistics —
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 0.973/1.024/1.123/0.057 ms
mysqlasrvadmin@MySQLsrv:~$

```

7. Create firewall rules in azure networking allowing traffic over port 3306 for both client and server:

Priority	Name	Port	Protocol	Source	Destination	Action	
100	AllowAnyMySQLOutbound	3306	TCP	Any	10.0.1.4	Allow	...
65000	AllowVnetOutBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow	...
65001	AllowInternetOutBound	Any	Any	Any	Internet	Allow	...
65500	DenyAllInBound	Any	Any	Any	Any	Deny	...

Inbound port rules Outbound port rules Application security groups Load balancing

Network security group **mySQLns-ns-g** (attached to network interface: **mysqlnw71**)
Impacts 0 subnets, 1 network interfaces

Add inbound port rule

Priority	Name	Port	Protocol	Source	Destination	Action	
300	SSH	22	TCP	Any	Any	Allow	...
310	AllowAnyMySQLInbound	3306	TCP	Any	Any	Allow	...
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow	...
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow	...
65500	DenyAllInBound	Any	Any	Any	Any	Deny	...

Firewall showing that you've allowed port commonly used by MySQL along with SSH^

8. Create remote user account with root privileges for clients:

```
mysql> CREATE USER 'lampy'@'10.0.0.4';
ERROR 1819 (HY000): Your password does not satisfy the current policy requirements
mysql> CREATE USER 'lampy'@'10.0.0.4' IDENTIFIED BY 'post5$^0986GREEN';
Query OK, 0 rows affected (0.03 sec)

mysql> GRANT ALL ON *.* TO 'lampy'@'10.0.0.4' WITH GRANT OPTION;
Query OK, 0 rows affected (0.04 sec)

mysql> show grants for 'lampy'@'10.0.0.4';
```

9. Confirm account creation and priviledges:

```
mysql> show grants for 'lampy'@'10.0.0.4';
```

```
| GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, SHUTDOWN, PROCES
SS, FILE, REFERENCES, INDEX, ALTER, SHOW DATABASES, SUPER, CREATE TEMPORARY T
ABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VI
EW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER, CR
EATE TABLESPACE, CREATE ROLE, DROP ROLE ON *.* TO `lampy`@`10.0.0.4` WITH GRA
NT OPTION
```

```
| GRANT APPLICATION_PASSWORD_ADMIN,AUDIT_ABORT_EXEMPT,AUDIT_ADMIN,AUTHENTICAT
ION_POLICY_ADMIN,BACKUP_ADMIN,BINLOG_ADMIN,BINLOG_ENCRYPTION_ADMIN,CLONE_ADMIN,
CONNECTION_ADMIN,ENCRYPTION_KEY_ADMIN,FIREWALL_EXEMPT,FLUSH_OPTIMIZER_COSTS
,FLUSH_STATUS,FLUSH_TABLES,FLUSH_USER_RESOURCES,GROUP_REPLICATION_ADMIN,GROUP
_REPLICATION_STREAM,INNODB_REDO_LOG_ARCHIVE,INNODB_REDO_LOG_ENABLE,PASSWORDLE
SS_USER_ADMIN,PERSIST_RO_VARIABLES_ADMIN,REPLICATION_APPLIER,REPLICATION_SLAV
E_ADMIN,RESOURCE_GROUP_ADMIN,RESOURCE_GROUP_USER,ROLE_ADMIN,SENSITIVE_VARIABLE
ES_OBSERVER,SERVICE_CONNECTION_ADMIN,SESSION_VARIABLES_ADMIN,SET_USER_ID,SHOW
_ROUTINE,SYSTEM_USER,SYSTEM_VARIABLES_ADMIN,TABLE_ENCRYPTION_ADMIN,XA_RECOVER
_ADMIN ON *.* TO `lampy`@`10.0.0.4` WITH GRANT OPTION |
```

10. Install mysql-client on LAMP server:

```
lampyadmin@LAMPy:~$ sudo apt install mysql-client
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
mysql-client is already the newest version (8.0.31-0ubuntu0.22.04.1).
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
```

11. Confirm remote account functionality by connecting from port 3306 from LAMP server:

```
lampyadmin@LAMPy:~$ mysql -h 10.0.1.4 -u lampy -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 8.0.31 MySQL Community Server - GPL

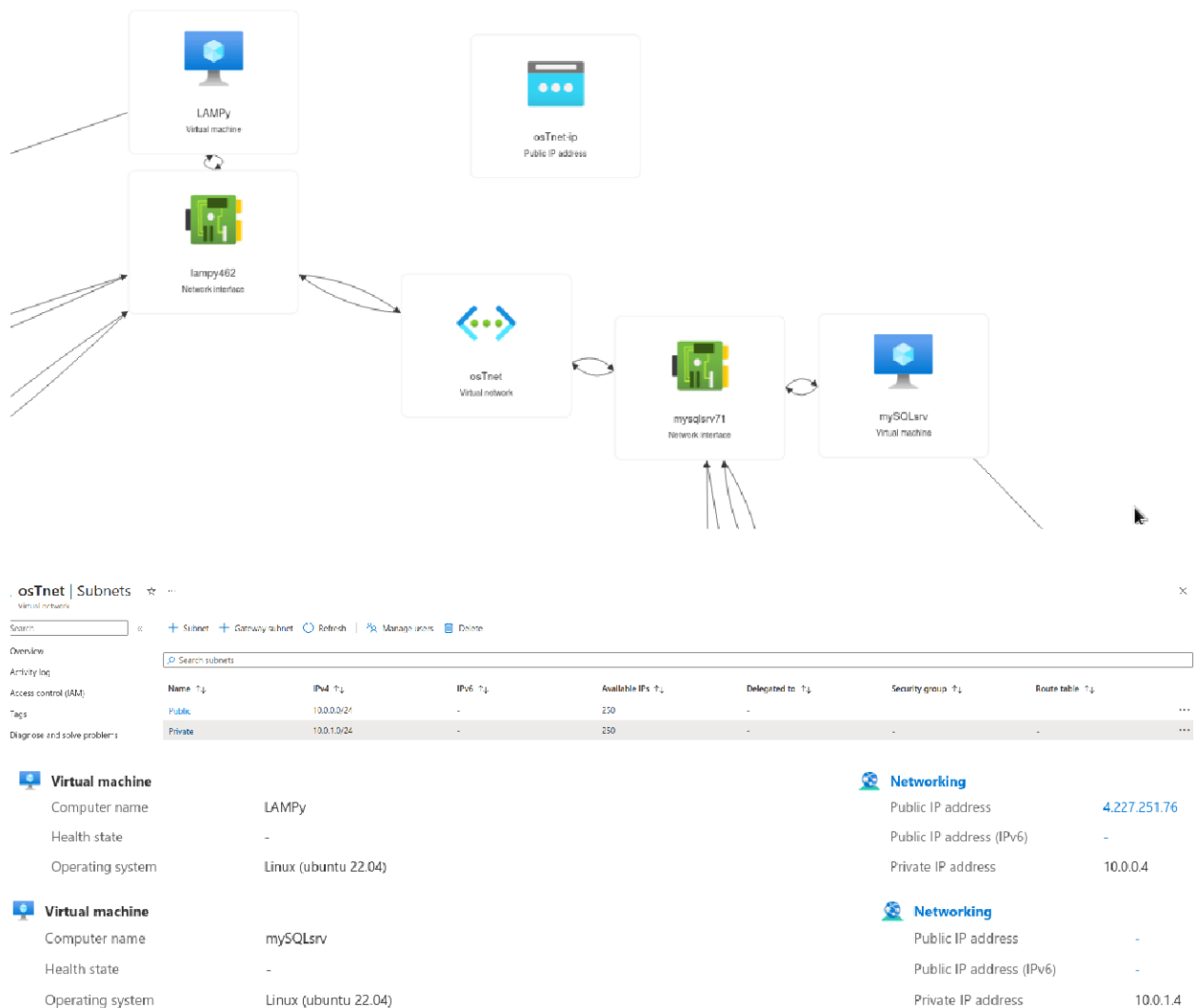
Copyright (c) 2000, 2022, Oracle and/or its affiliates.

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>
mysql> exit
Bye
```

Successfully show that you can communicate with the VM from the LAMP VM. ^



Show that this VM is on a different network than the LAMP server.

Command History —>

Command History:

LAMP server:

users

cd

ls

apt-get update

sudo apt-get update

sudo apt-get upgrade

lsb_release -a command

lsb_release command

lsb_release -a

```
sudo ufw allow ssh
sudo ufw allow 80
sudo ufw allow 443
sudo ufw enable
sudo apt install apache2
sudo systemctl status apache2
exit
ping -c4 10.0.1.4
mysql root@10.0.1.4 -p
sudo apt install mysql-client
mysql root@10.0.1.4 -p
systemctl status mysqld
mysql
user
users
mysqladmin -u root -p status
service mysqld start
mysqladmin -u root --skip-password
mysqladmin -u root --skip-password status
ls /etc/
ls /etc/ | grep my
ls /etc/mysql/
cat /etc/mysql/my.cnf
less /var/log/mysql
mysql --host=10.0.1.4 --user=root
mysql --host=10.0.1.4 --user=root -proot
mysql --host=10.0.1.4 --user=root
mysql --host=10.0.1.4/16 --user=root
mysql --host=10.0.1.4 --user=root
mysql --host=10.0.1.4 --user=root -p
mysql --host=10.0.1.4 --user=root -p -P 3306
users
history | less
history
mysql -h 10.0.1.4 -u lampy -p
```

MySQL Server SQL History:

```
_HiStOrY_V2_
help;
quit
exit
use\040mysql
show\040grants\040for\040root
show\040grants\040for\040'root'@'localhost';
```

```
show\040grants
show\040grants\040for\040'root'@'%';
show\040grants\040for\040root@'%';
show\040grants\040for\040root@%;
GRANT\040ALL\040ON\040*.*\040TO\040'root'@'10.0.0.4'\040WITH\040GRANT\040OPTION;
show\040grants\040for\040'root'@'10.0.0.4'
show\040grants\040for\040'root'@'10.0.0.4';
exit
GRANT\040ALL\040ON\040*.*\040TO\040'lampy'@'10.0.0.4'\040WITH\040GRANT\040OPTIO
N;
help
CREATE\040USER\040'lampy'@'10.0.0.4'
CREATE\040USER\040'lampy'@'10.0.0.4';
GRANT\040ALL\040ON\040*.*\040TO\040'lampy'@'10.0.0.4'\040WITH\040GRANT\040OPTIO
N;
show\040grants\040for\040'lampy'@'10.0.0.4';
Exit
```

Local History:

```
mv Downloads/osTicket\ project\ ssh\ keys.zip Desktop/School/
cd Desktop/School
mkdir COSN299
mv osTicket\ project\ ssh\ keys.zip COSN299
ls COSN299
cd COSN299
7z osTicket\ project\ ssh\ keys.zip
man 7z
7z -x osTicket\ project\ ssh\ keys.zip
7z x osTicket\ project\ ssh\ keys.zip
file __MACOSX
ls __MACOSX
rm __MACOSX
rmdir __MACOSX
rm -r __MACOSX
less osTicket\ project\ ssh\ keys
ls osTicket\ project\ ssh\ keys
ls osTicket\ project\ ssh\ keys\untitled\ folder
rm osTicket\ project\ ssh\ keys.zip
cd osTicket\ project\ ssh\ keys
rm untitled\ folder
rmdir untitled\ folder
man ssh
mv * ~
cd ~
```



```
ls -a
ls .ssh/known_hosts
mv lampykey.pem .ssh/known_hosts
mv mySQLsrv_key.pem .ssh/known_hosts
cd .ssh
cat known_hosts
cat known_hosts.old
mv ~/osTicket\ project\ ssh\ keys .
cat osTicket\ project\ ssh\ keys/osTicket\ project\ ssh\ keys/mySQLsrv_key.pem
cat osTicket\ project\ ssh\ keys/osTicket\ project\ ssh\ keys/lampykey.pem
vim known_hosts
less known_hosts
ls
cd ..
ssh -i ~/.ssh/known_hosts mysqlasrvadmin@172.174.81.60
ssh -i ~/.ssh/mySQLsrv_key.pem mysqlasrvadmin@172.174.81.60
chmod 400 .ssh/lampykey.pem .ssh/mySQLsrv_key.pem
ls -l .ssh
ssh -i ~/.ssh/mySQLsrv_key.pem mysqlasrvadmin@172.174.81.60
```