

# Step4: DBサーバー構築 (MariaDB)

# 1. 目的

---

- DBServer に MariaDB を導入し、サービスを稼働させる
- AppServer から接続して DB連携を確認する
- 実務を意識し、セキュリティ設定やユーザー権限も適切に設定する

## 2. MariaDB インストール (DBServer側)

- SSHログイン

```
ssh -p 2222 admin@127.0.0.1
```

- インストール

```
sudo dnf install -y mariadb-server
```

- サービス起動と自動起動設定

```
sudo systemctl enable --now mariadb
```

```
systemctl status mariadb
```

```
perl-Math-Complex-1.39-481.1.el9_6.noarch
perl-NDBM_File-1.15-481.1.el9_6.x86_64
perl-POSIX-1.94-481.1.el9_6.x86_64
perl-Pod-Escapes-1:1.07-460.el9.noarch
perl-Pod-Simple-1:3.42-4.el9.noarch
perl-Scalar-List-Utils-4:1.56-462.el9.x86_64
perl-Socket-4:2.031-4.el9.x86_64
perl-Symbol-1.08-481.1.el9_6.noarch
perl-Term-ANSIColor-5.01-461.el9.noarch
perl-Text-ParseWords-3.30-460.el9.noarch
perl-Time-Local-2:1.300-7.el9.noarch
perl-base-2.27-481.1.el9_6.noarch
perl-if-0.60.800-481.1.el9_6.noarch
perl-libnet-3.13-4.el9.noarch
perl-mro-1.23-481.1.el9_6.x86_64
perl-overloading-0.02-481.1.el9_6.noarch
perl-podlators-1:4.14-460.el9.noarch
perl-vars-1.05-481.1.el9_6.noarch
python3-audit-3.1.5-4.el9.x86_64
python3-libsemanage-3.6-5.el9_6.x86_64
python3-setuptools-4.4.4-1.el9.x86_64
perl-Mozilla-CSS-20200520-8.el9.noarch
perl-Net-SSLeay-1.94-1.el9.x86_64
perl-PathTools-3.78-461.el9.x86_64
perl-Pod-Perldoc-3.28.01-461.el9.noarch
perl-Pod-Usage-4:2.01-4.el9.noarch
perl-SelectSaver-1.02-481.1.el9_6.noarch
perl-Storable-1:3.21-460.el9.x86_64
perl-Sys-Hostname-1.23-481.1.el9_6.x86_64
perl-Term-Cap-1.17-460.el9.noarch
perl-Text-Tabs+Wrap-2013.0523-460.el9.noarch
perl-URI-5.09-3.el9.noarch
perl-constant-1.33-461.el9.noarch
perl-interpreter-4:5.32.1-481.1.el9_6.x86_64
perl-libs-4:5.32.1-481.1.el9_6.x86_64
perl-overload-1.31-481.1.el9_6.noarch
perl-parent-1:0.238-460.el9.noarch
perl-subst-1.03-481.1.el9_6.noarch
policycoreutils-python-utils-3.6-2.1.el9.noarch
python3-distro-1.5.0-7.el9.noarch
python3-policycoreutils-3.6-2.1.el9.noarch
python3-setuptools-53.0.0-13.el9_6.1.noarch
complete!
```

```
Sep 05 18:22:50 vbox mariadb-prepare-db-dir[3592]: The second is mysql@localhost, it has no password either, but
Sep 05 18:22:50 vbox mariadb-prepare-db-dir[3592]: you need to be the system 'mysql' user to connect.
Sep 05 18:22:50 vbox mariadb-prepare-db-dir[3592]: After connecting you can set the password, if you would need to be
Sep 05 18:22:50 vbox mariadb-prepare-db-dir[3592]: able to connect as any of these users with a password and without sudo
Sep 05 18:22:50 vbox mariadb-prepare-db-dir[3592]: See the MariaDB Knowledgebase at https://mariadb.com/kb
Sep 05 18:22:50 vbox mariadb-prepare-db-dir[3592]: Please report any problems at https://mariadb.org/jira
Sep 05 18:22:50 vbox mariadb-prepare-db-dir[3592]: The latest information about MariaDB is available at https://mariadb.org/.
Sep 05 18:22:50 vbox mariadb-prepare-db-dir[3592]: Consider joining MariaDB's strong and vibrant community:
Sep 05 18:22:50 vbox mariadb-prepare-db-dir[3592]: https://mariadb.org/get-involved/
Sep 05 18:22:50 vbox systemd[1]: Started MariaDB 10.5 database server.
```

### 3. 初期設定（セキュリティ強化）

`sudo mysql_secure_installation`

推奨設定例:

- rootパスワード設定: Yes
- 匿名ユーザー削除: Yes
- リモートrootログイン禁止: Yes
- テストDB削除: Yes
- 権限テーブル再読み込み: Yes

```
Remove test database and access to it? [Y/n] 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? [Y/n] Y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.
```

## 4. DB作成とユーザー権限

---

MariaDBにログイン:

```
sudo mysql -u root -p
```

アプリ用DBとユーザーを作成（例: appdb, appuser）:

```
CREATE DATABASE appdb;
```

```
CREATE USER 'appuser'@'%' IDENTIFIED BY 'goodPasw0rds'; (例)
```

```
GRANT ALL PRIVILEGES ON appdb.* TO 'appuser'@'%';
```

```
FLUSH PRIVILEGES;
```

```
EXIT;
```

## 5. Firewall設定 (DBServer側)

---

AppServerからの接続を許可します。

```
sudo firewall-cmd --permanent --add-service=mysql
```

```
sudo firewall-cmd --reload
```

```
[redacted]@vbox ~]$ sudo firewall-cmd --permanent --add-service=mysql
success
[redacted]@vbox ~]$
[redacted]@vbox ~]$
[redacted]@vbox ~]$ sudo firewall-cmd --reload
success
```

## 6. 接続確認（AppServer側）

---

- AppServerにMySQLクライアントを導入

```
sudo dnf install -y mariadb
```

- DBServerへ接続確認

```
mysql -h 10.0.2.16 -u appuser -p appdb
```

（10.0.2.16 は DBServer のNAT内IP。実際は `ip a` で確認）

- 確認用テーブルを作成してみる

```
CREATE TABLE test (id INT PRIMARY KEY, name VARCHAR(50));
```

```
INSERT INTO test VALUES (1, 'Hello DB');
```

```
SELECT * FROM test;
```

# 補足: VM間接続のためのネットワーク再設計

---

## 1. 目的

- ・ NAT構成のみでは VM 間通信ができないため、AppServer ↔ DBServer 間の直接接続を可能にする。
- ・ VirtualBox の **Host-Onlyネットワーク** を追加し、Web→DB接続を実現する。



# 補足: VM間接続のためのネットワーク再設計

---

## 2. VirtualBox側の設定

1. 各VM (AppServer, DBServer) を停止。
2. 設定 → ネットワーク → **アダプタ2**を有効化。
  - 接続先: Host-Only Adapter
  - 名前: vboxnet0 (デフォルト)

# 補足: VM間接続のためのネットワーク再設計

---

## 3. IPアドレス設定 (nmcli, 各VM内)

### AppServer

```
sudo nmcli con add type ethernet ifname enp0s8 con-name hostonly ¥  
ipv4.method manual ipv4.addresses 192.168.56.101/24  
sudo nmcli con up hostonly
```

### DBServer

```
sudo nmcli con add type ethernet ifname enp0s8 con-name hostonly ¥  
ipv4.method manual ipv4.addresses 192.168.56.102/24  
sudo nmcli con up hostonly
```

# 補足: VM間接続のためのネットワーク再設計

---

## 4. MariaDBのbind設定 (DBServer)

MariaDBをHost-Only IPで待ち受けるように変更。

```
sudo vi /etc/my.cnf.d/mariadb-server.cnf
```

以下を [mysqld] セクションに追記/修正：

```
bind-address=0.0.0.0
```

再起動：

```
sudo systemctl restart mariadb
```

# 補足: VM間接続のためのネットワーク再設計

## 5. Firewall設定 (DBServer)

Host-OnlyネットワークからのMySQL接続を許可。

```
sudo firewall-cmd --permanent --add-service=mysql
```

```
sudo firewall-cmd --reload
```

```
[redacted]@vbox ~]$ sudo systemctl restart mariadb
[redacted]@vbox ~]$ sudo firewall-cmd --permanent --add-service=mysql
Warning: ALREADY_ENABLED: mysql
success
[redacted]@vbox ~]$
[redacted]@vbox ~]$
[redacted]@vbox ~]$ sudo firewall-cmd --reload
success
```

# 補足: VM間接続のためのネットワーク再設計

## 6. DBユーザー権限再設定

Host-Only用のユーザーを許可。

rootでMariaDBログイン

```
sudo mysql -u root -p
```

```
GRANT ALL PRIVILEGES ON appdb.* TO  
'appuser'@'192.168.56.%' IDENTIFIED BY 'goodPassw0rds';  
FLUSH PRIVILEGES;
```

```
MariaDB [(none)]>  
MariaDB [(none)]> GRANT ALL PRIVILEGES ON appdb.* TO ' ' @' .%' IDENTIFIED BY ' '; FLUSH PRI  
VILEGES;  
Query OK, 0 rows affected (0.003 sec)  
  
Query OK, 0 rows affected (0.001 sec)  
  
MariaDB [(none)]> exit  
Bye
```

# 補足: VM間接続のためのネットワーク再設計

## 7. 接続確認 (AppServer側)

### 1. 到達性確認

ping 192.168.56.102

### 2. DB接続テスト

mysql -h 192.168.56.102 -u appuser -p appdb

### 3. テストテーブル作成

```
CREATE TABLE test (id INT PRIMARY KEY, name VARCHAR(50));  
INSERT INTO test VALUES (1, 'Hello DB');  
SELECT * FROM test;
```

```
MariaDB [appdb]> INSERT INTO test VALUES (1, 'Hello DB');  
Query OK, 1 row affected (0.003 sec)  
  
MariaDB [appdb]> SELECT * FROM test;  
+----+-----+  
| id | name  |  
+----+-----+  
|  1 | Hello DB |  
+----+-----+  
1 row in set (0.001 sec)
```