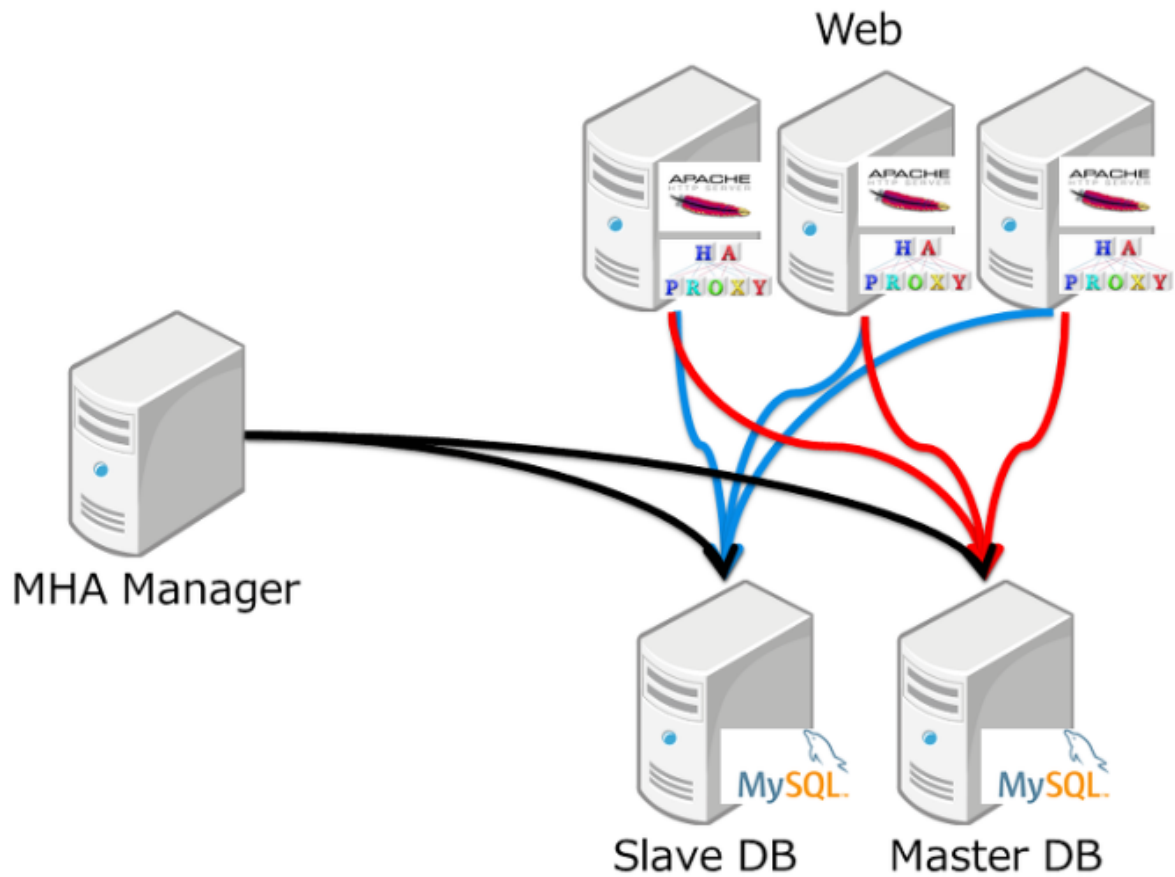


Database: Linux

- [Database Replication](#)

Database Replication

MHA(Master High Availability)は、Master DBが障害発生した場合、自動的にSlave DBにフェイルオーバーを行う。



MHAはMHA managerが障害発生した場合、自動的にMasterDBをSlaveDBにフェイルオーバーさせる。

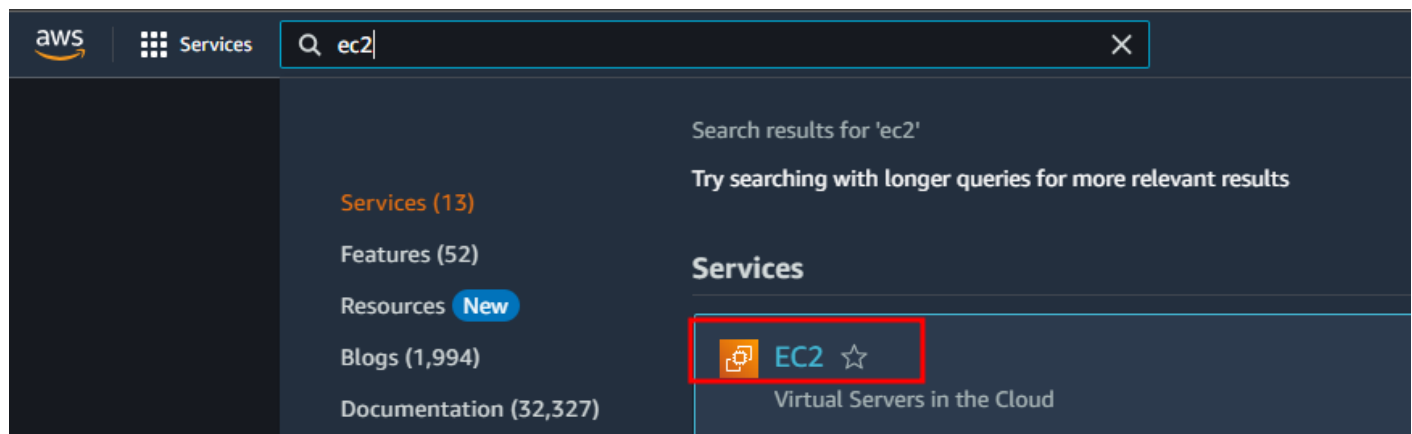
2台のDBのうち1台はMaster、もう1台はSlave(replication)DBとして構成される。

構成要素

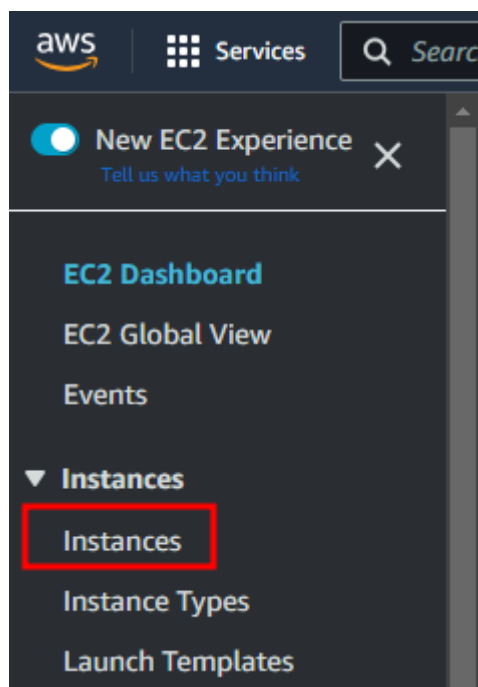
- 2台のAWS EC2 インスタンス (Cloud上のVirtual Machine)
- 1台 x86_64 CPU インスタンス または 1台 ARM CPU インスタンス
- DB-Master インスタンス (172.31.33.8) に RHEL 9.2
- DB-Slave インスタンス (172.31.23.223) に RHEL 9.2

EC2 インスタンス

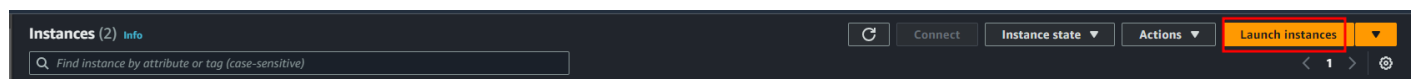
aws 서비스 검색 EC2 서비스 찾기



aws 콘솔 'instances' 메뉴 찾기



aws 콘솔 'Launch Instances' 버튼 찾기



aws 콘솔, CPU 옵션 선택

Name and tags [Info](#)

Name

DB-Master

[Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Recents

Quick Start

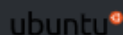
Amazon
Linux



macOS



Ubuntu



Windows



Red Hat



SUSE Li



[Browse more AMIs](#)

Including AMIs from
AWS, Marketplace and
the Community

Amazon Machine Image (AMI)

Red Hat Enterprise Linux 9 (HVM), SSD Volume Type

ami-004b403708f61ecd8 (64-bit (x86)) / ami-06d71950b140f27c5 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible ▼

Description

Provided by Red Hat, Inc.

Architecture

64-bit (x86) ▼

AMI ID

ami-004b403708f61ecd8

Verified provider

Slave DB ARM

Name and tags [Info](#)


Name

DB-Slave

[Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

 Search our full catalog including 1000s of application and OS images

Recents

Quick Start

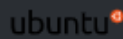
Amazon
Linux



macOS



Ubuntu



Windows



Red Hat



SUSE Li



[Browse more AMIs](#)

Including AMIs from
AWS, Marketplace and
the Community

Amazon Machine Image (AMI)

Red Hat Enterprise Linux 9 (HVM), SSD Volume Type

Free tier eligible ▼

ami-004b403708f61ecd8 (64-bit (x86)) / ami-06d71950b140f27c5 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Provided by Red Hat, Inc.

Architecture

64-bit (Arm)



AMI ID

ami-06d71950b140f27c5

Verified provider

0000 0000 00 00, 0000 00 0000 0000 0000

▼ Instance type [Info](#)

Instance type

t4g.nano

Family: t4g 2 vCPU 0.5 GiB Memory Current generation: true

On-Demand Linux pricing: 0.0052 USD per Hour

On-Demand SUSE pricing: 0.0052 USD per Hour

☒ All generations

Compare instance types

► **Key pair (login)** [Info](#)

▼ **Network settings** [Info](#)

Edit

Network Info

vpc-05d0a325871436a03

Subnet Info

No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)

Enable

Firewall (security groups) Info

- Create security group

- Select existing security group

Common security groups [Info](#)

Select security groups

C

Compare security group rules

▼ **Configure storage** [Info](#)

Advanced

1x

10

GiB

gp2

Root volume (Not encrypted)

AWS Firewall

Block

이제 생성한 키 쌍을 SSH 접속에 사용할 수 있습니다.

Create key pair

Key pair name
Key pairs allow you to connect to your instance securely.

The name can include upto 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

☒ RSA
RSA encrypted private and public key pair

☐ ED25519
ED25519 encrypted private and public key pair

Private key file format

☒ .pem
For use with OpenSSH

☐ .ppk
For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#)

Cancel

Create key pair

이제 생성한 키 쌍을 SSH 접속에 사용할 수 있습니다.

<input type="checkbox"/>	DB-Slave	i-0a7fa2b2f131ea44b	Running		t4g.small
<input type="checkbox"/>	DB-Master	i-097725e0ede023162	Running		t2.micro

이제 생성한 키 쌍을 SSH 접속에 사용할 수 있습니다.

1 DB-Master × 2 DB-Slave × +

[root@ip-172-31-33-8 ~]#

OS DB CPU DB DB

DB OS

```
[root@ip-172-31-33-8 ~]# cat /etc/os-release
```

NAME="Red Hat Enterprise Linux"

VERSION="9.2 (Plow)"

DB CPU DB

```
[root@ip-172-31-33-8 ~]# lscpu
```

Architecture: x86_64

CPU op-mode(s): 32-bit, 64-bit

Address sizes: 46 bits physical, 48 bits virtual

Byte Order: Little Endian

CPU(s): 1

On-line CPU(s) list: 0

Vendor ID: GenuineIntel

BIOS Vendor ID: Intel

Model name: Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz

Slave DB OS

```
[root@ip-172-31-23-223 ~]# cat /etc/os-release
```

NAME="Red Hat Enterprise Linux"

VERSION="9.2 (Plow)"

Slave DB CPU DB

```
[root@ip-172-31-23-223 ~]# lscpu
```

Architecture: aarch64

CPU op-mode(s): 32-bit, 64-bit

Byte Order: Little Endian

CPU(s): 2

On-line CPU(s) list: 0,1

Vendor ID: ARM

BIOS Vendor ID: AWS

Model name: Neoverse-N1

BIOS Model name: AWS Graviton2

MariaDB v15.1

```
mariaDB[ ] Monty[ ] mysql [ ] [ ] MontyProgram AB[ ] [ ] [ ] mariaDB[ ] [ ] [ ].  
[ ] mariaDB[ ] mysql[ ] [ ] [ ] [ ]
```

MariaDB   Master  Slave  .   Master  Slave   mariaDB    

```
# 1.1.1.1 1.1.1.1
sudo dnf update -y

# MariaDB 1.1.1.1
sudo dnf -y install mariadb-*

# MariaDB 1.1.1.1
sudo systemctl enable mariadb
sudo systemctl start mariadb

# MasterDB 1.1.1.1 MariaDB 1.1.1.1
[root@ip-172-31-33-8 ~]# mariadb --version
mariadb Ver 15.1 Distrib 10.5.16-MariaDB, for Linux (x86_64) using EditLine wrapper

# SlaveDB 1.1.1.1 MariaDB 1.1.1.1
[root@ip-172-31-23-223 ~]# mariadb --version
mariadb Ver 15.1 Distrib 10.5.16-MariaDB, for Linux (aarch64) using EditLine wrapper

# root 1.1.1.1 MariaDB 1.1.1.1 1.1.1.1 1.1.1.1
sudo su - root
mysql_secure_installation

# 1.1.1.1 1.1.1.1
Enter current password for root (enter for none):
OK, successfully used password, moving on...

Remove anonymous users? [Y/n] y
... Success!

Disallow root login remotely? [Y/n] n
... skipping.
```

Remove test database and access to it? [Y/n] y

- Dropping test database...

... Success!

Removing privileges on test database...

... Success!

Reload privilege tables now? [Y/n] y

... Success!

Cleaning up...

All done!

Thanks for using MariaDB!

MariaDB

MariaDB

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

log-bin=mysql-bin

server-id=1

MariaDBMS

mysql -u root

DB

MariaDB [(none)]> create database testDB;

DB

MariaDB [(none)]> show databases;

+-----+

| Database |

+-----+

| information_schema |

| mysql |

| performance_schema |

```
| testDB          |
```

```
+-----+
```

```
4 rows in set (0.000 sec)
```

```
# Master DB Privileges
```

```
MariaDB [mysql]> grant all privileges on testDB.* to 'Master'@'%' identified by '1';
```

```
Query OK, 0 rows affected (0.003 sec)
```

```
# SlaveDB replication
```

```
MariaDB [mysql]> grant replication slave on *.* to 'Slave'@'%' identified by '1';
```

```
Query OK, 0 rows affected (0.001 sec)
```

```
## DBMS mariadb
```

```
MariaDB [mysql]> exit
```

```
systemctl restart mariadb
```

```
# SlaveDB Position ID
```

```
mysql -u root -p -e "show master status"
```

```
+-----+-----+-----+-----+
```

```
| File          | Position | Binlog_Do_DB | Binlog_Ignore_DB |
```

```
+-----+-----+-----+-----+
```

```
| mysql-bin.000003 | 342 | | |
```

```
+-----+-----+-----+-----+
```

```
# DB Backup
```

```
mysqldump -u root --all-databases > backup.sql
```

```
# SCP DB Backup SlaveDB
```

```
scp backup.sql root@172.31.23.223:/root/
```

SlaveDB

```
# MariaDB
```

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

```
server-id=2
```

```
replicate-do-db='testDB'
```

```
# MasterDB Backup [] []
```

```
sudo su - root
```

```
mysql -u root< backup.sql
```

```
# MariaDBMS []
```

```
mysql -u root
```

```
# MasterDB[] [] DB []
```

```
MariaDB [(none)]> show databases;
```

```
+-----+
```

```
| Database      |
```

```
+-----+
```

```
| testDB        |
```

```
| information_schema |
```

```
| mysql          |
```

```
| performance_schema |
```

```
+-----+
```

```
4 rows in set (0.007 sec)
```

```
# Slave []
```

```
MariaDB [(none)]> change master to master_host='172.31.33.8',
```

```
-> master_user='Slave',
```

```
-> master_password='1',
```

```
-> master_log_file='mysql-bin.000003',
```

```
-> master_log_pos=342;
```

```
Query OK, 0 rows affected (0.009 sec)
```

```
# MariaDB []
```

```
systemctl restart mariadb
```

[] Master[] Slave [][][] [] [] []. [] [] [] [] 'testDB'[] [] [] [] [] [] [] [] [] [] [] []

```
## MasterDB[] []
```

```
[root@ip-172-31-33-8 ~]# mysql -u root
```

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
```

```
Your MariaDB connection id is 4
```

```
Server version: 10.5.16-MariaDB-log MariaDB Server
```

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> use testDB;

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

Database changed

MariaDB [testDB]> show tables;

```
+-----+
| Tables_in_testDB |
+-----+
| test              |
+-----+
1 row in set (0.000 sec)
```

test2 ☐☐☐☐

MariaDB [testDB]> create table test2 (a int);

Query OK, 0 rows affected (0.009 sec)

MariaDB [testDB]> show tables;

```
+-----+
| Tables_in_testDB |
+-----+
| test              |
| test2             |
+-----+
2 rows in set (0.001 sec)
```

test3 ☐☐☐☐☐☐

MariaDB [testDB]> create table test3 (a int);

Query OK, 0 rows affected (0.007 sec)

MariaDB [testDB]> show tables;

```
+-----+
| Tables_in_testDB |
+-----+
| test              |
| test2             |
```

```
| test3      |
+-----+
3 rows in set (0.000 sec)
```

SlaveDB Replication

```
## SlaveDB
[root@ip-172-31-23-223 ~]# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 6
Server version: 10.5.16-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

## testDB
MariaDB [(none)]> use testDB;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
# MasterDB test2
MariaDB [testDB]> show tables;
+-----+
| Tables_in_testDB |
+-----+
| test              |
| test2             |
+-----+
2 rows in set (0.000 sec)

# MasterDB test3
MariaDB [testDB]> show tables;
+-----+
| Tables_in_testDB |
+-----+
| test              |
```

```
| test2      |
| test3      |
+-----+
3 rows in set (0.000 sec)
```

MHA Manager 環境 構築 MHA MySQL DB Replication 構築

環境 構築 MySQL v5.7 環境 構築, MHA manager, MasterDB 構築 SlaveDB 構築 構築 構築. MHA manager 構築 構築 au 構築 構築 構築

- 3 構築 AWS EC2 構築 (Cloud 構築 Virtual Machine)
- 2 構築(MHA+Master) 構築 x86_64 CPU 構築 構築 構築 1(Slave) 構築 ARM CPU 構築 構築 構築
- MHA-Manager 構築 (172.31.45.57) 構築 RHEL 9.2
- DB-Master 構築 (172.31.35.181) 構築 RHEL 9.2
- DB-Slave 構築 (172.31.25.160) 構築 RHEL 9.2

EC2 構築 構築 構築 構築 構築 構築 構築 OS 構築 CPU Architecture 構築 構築 構築.

```
## MHA OS 構築 構築 構築
$ cat /etc/os-release
NAME="Red Hat Enterprise Linux"
VERSION="9.2 (Plow)"

$ lscpu
Architecture:      x86_64

## 構築 DB OS 構築 構築 構築
$ cat /etc/os-release
NAME="Red Hat Enterprise Linux"
VERSION="9.2 (Plow)"

$ lscpu
Architecture:      x86_64

## Slave DB OS 構築 構築 構築
$ cat /etc/os-release
NAME="Red Hat Enterprise Linux"
```

```
VERSION="9.2 (Plow)"
```

```
$ lscpu
```

```
Architecture:      aarch64
```

DB 环境搭建

环境搭建 MySQL v5.7 环境

MySQL 环境搭建 rpm 环境搭建, 环境搭建

```
# GPG 导入
```

```
sudo rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2022
```

```
# rpm 安装
```

```
wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
```

```
# mysql 安装
```

```
yum localinstall mysql57-community-release-el7-11.noarch.rpm
```

```
yum repolist enabled | grep "mysql.-community."
```

```
yum install mysql-community-server -y
```

```
# mysqld 启动
```

```
systemctl start mysqld
```

```
systemctl status mysqld
```