# Install Tiebreaker 1.5 or 1.4

## ONTAP MetroCluster

NetApp
March 29, 2024

# Table of Contents

# Install Tiebreaker 1.5 or 1.4

## Configure admin access to ONTAP API and SSH

You can configure admin access to ONTAP API and SSH.

**Steps**

1. Create an admin user with ONTAP API access: `security login create -user-or-group-name mcctb -application ontapi -authentication-method password`
2. Create an admin user with SSH access: `security login create -user-or-group-name mcctb -application ssh -authentication-method password`
3. Verify that the new admin users are created: `security login show`
4. Repeat these steps on the partner cluster.

   Administrator authentication and RBAC is implemented.

## Installing MetroCluster Tiebreaker dependencies

**Related information**

Depending on your host Linux operating system, you must install a MySQL or MariaDB server before installing or upgrading the Tiebreaker software.

**Steps**

1. Install JRE.

   Install JRE

2. Install and configure Vault.

   Install and configure Vault

3. Install MySQL or MariaDB server:

| If the Linux host is | Then… |
|---|---|
| Red Hat Enterprise Linux 7/CentOS 7 | Install MySQL<br>Installing MySQL Server 5.5.30 or later and 5.6.x versions on Red Hat Enterprise Linux 7 or CentOS 7 |
| Red Hat Enterprise Linux 8 | Install MariaDB<br>Installing MariaDB server on Red Hat Enterprise Linux 8 |

## Install JRE

You must install JRE on your host system before installing or upgrading the Tiebreaker software. For systems running Tiebreaker 1.4 and earlier, run JRE 8. For systems running Tiebreaker 1.5 and later, run OpenJDK 17, 18, or 19.

> The outputs in the following example show JRE 1.8.0. (JRE 8).

**Steps**

1. Log in as a "root" user or a sudo user that can change to advanced privilege mode.

   ```
   login as: root
   root@mcctb's password:
   Last login: Fri Jan  8 21:33:00 2017 from host.domain.com
   ```

2. Check for available JRE versions:

   ```
   yum search openjdk
   ```

3. Install an appropriate JRE version for the version of Tiebreaker you are installing:

   ```
   yum install java-<version>-openjdk.x86_64
   ```

   ```
   [root@mcctb ~]# yum install java-1.8.0-openjdk.x86_64
   Loaded plugins: fastestmirror, langpacks
   Loading mirror speeds from cached hostfile
   ... shortened....
   Dependencies Resolved

   =======================================================================
   Package               Arch   Version                Repository    Size
   =======================================================================
   Installing:
    java-1.8.0-openjdk  x86_64  1:1.8.0.144-0.b01.el7_4 updates      238 k
    ..
    ..
   Transaction Summary
   =======================================================================
   Install  1 Package  (+ 4 Dependent packages)

   Total download size: 34 M
   Is this ok [y/d/N]: y

   Installed:
   java-1.8.0-openjdk.x86_64 1:1.8.0.144-0.b01.el7_4
   Complete!
   ```

## Install and configure Vault

If you do not have or want to use the local Vault server, you must install Vault. You can refer to this standard procedure for installing Vault, or refer to the Hashicorp installation instructions for alternative guidelines.

> If you have a Vault server in your network, you can configure the MetroCluster Tiebreaker host to use that Vault installation. If you do this, you do not need to install Vault on the host.

**Steps**

1. Navigate to the `/bin` directory:

```
[root@mcctb] cd /bin
```

2. Download the Vault zip file.

```
[root@mcctb /bin]#  curl -sO
https://releases.hashicorp.com/vault/1.12.2/vault_1.12.2_linux_amd64.zip
```

3. Unzip the Vault file.

```
[root@mcctb /bin]# unzip vault_1.12.2_linux_amd64.zip
Archive:  vault_1.12.2_linux_amd64.zip
  inflating: vault
```

4. Verify the installation.

```
[root@mcctb /bin]# vault -version
Vault v1.12.2 (415e1fe3118eebd5df6cb60d13defdc01aa17b03), built 2022-11-
23T12:53:46Z
```

5. Navigate to the `/root` directory:

```
[root@mcctb /bin] cd /root
```

6. Create a Vault configuration file under the `/root` directory.

   At the `[root@mcctb ~]` prompt, copy and run the following command to create the `config.hcl` file:

```
# cat > config.hcl << EOF
 storage "file" {
  address = "127.0.0.1:8500"
  path    = "/mcctb_vdata/data"
 }
 listener "tcp" {
   address     = "127.0.0.1:8200"
   tls_disable = 1
 }
EOF
```

7. Start the Vault server:

```
[root@mcctb ~] vault server -config config.hcl &
```

8. Export the Vault address.

```
[root@mcctb ~]# export VAULT_ADDR="http://127.0.0.1:8200"
```

9. Initialize Vault.

```
[root@mcctb ~]# vault operator init
2022-12-15T14:57:22.113+0530 [INFO]  core: security barrier not
initialized
2022-12-15T14:57:22.113+0530 [INFO]  core: seal configuration missing,
not initialized
2022-12-15T14:57:22.114+0530 [INFO]  core: security barrier not
initialized
2022-12-15T14:57:22.116+0530 [INFO]  core: security barrier initialized:
stored=1 shares=5 threshold=3
2022-12-15T14:57:22.118+0530 [INFO]  core: post-unseal setup starting
2022-12-15T14:57:22.137+0530 [INFO]  core: loaded wrapping token key
2022-12-15T14:57:22.137+0530 [INFO]  core: Recorded vault version: vault
version=1.12.2 upgrade time="2022-12-15 09:27:22.137200412 +0000 UTC"
build date=2022-11-23T12:53:46Z
2022-12-15T14:57:22.137+0530 [INFO]  core: successfully setup plugin
catalog: plugin-directory=""
2022-12-15T14:57:22.137+0530 [INFO]  core: no mounts; adding default
mount table
2022-12-15T14:57:22.143+0530 [INFO]  core: successfully mounted backend:
type=cubbyhole version="" path=cubbyhole/
2022-12-15T14:57:22.144+0530 [INFO]  core: successfully mounted backend:
type=system version="" path=sys/
```

```
2022-12-15T14:57:22.144+0530 [INFO]  core: successfully mounted backend:
type=identity version="" path=identity/
2022-12-15T14:57:22.148+0530 [INFO]  core: successfully enabled
credential backend: type=token version="" path=token/ namespace="ID:
root. Path: "
2022-12-15T14:57:22.149+0530 [INFO]  rollback: starting rollback manager
2022-12-15T14:57:22.149+0530 [INFO]  core: restoring leases
2022-12-15T14:57:22.150+0530 [INFO]  expiration: lease restore complete
2022-12-15T14:57:22.150+0530 [INFO]  identity: entities restored
2022-12-15T14:57:22.150+0530 [INFO]  identity: groups restored
2022-12-15T14:57:22.151+0530 [INFO]  core: usage gauge collection is
disabled
2022-12-15T14:57:23.385+0530 [INFO]  core: post-unseal setup complete
2022-12-15T14:57:23.387+0530 [INFO]  core: root token generated
2022-12-15T14:57:23.387+0530 [INFO]  core: pre-seal teardown starting
2022-12-15T14:57:23.387+0530 [INFO]  rollback: stopping rollback manager
2022-12-15T14:57:23.387+0530 [INFO]  core: pre-seal teardown complete
Unseal Key 1: <unseal_key_1_id>
Unseal Key 2: <unseal_key_2_id>
Unseal Key 3: <unseal_key_3_id>
Unseal Key 4: <unseal_key_4_id>
Unseal Key 5: <unseal_key_5_id>

Initial Root Token: <initial_root_token_id>



Vault initialized with 5 key shares and a key threshold of 3. Please
securely
distribute the key shares printed above. When the Vault is re-sealed,
restarted, or stopped, you must supply at least 3 of these keys to
unseal it
before it can start servicing requests.

Vault does not store the generated root key. Without at least 3 keys to
reconstruct the root key, Vault will remain permanently sealed!

It is possible to generate new unseal keys, provided you have a quorum
of
existing unseal keys shares. See "vault operator rekey" for more
information.
```

ⓘ You must record and store the key IDs and initial root token in a secure location for use later in the procedure.

10. Export the Vault root token.

```
[root@mcctb ~]#  export VAULT_TOKEN="<initial_root_token_id>"
```

11. Unseal Vault by using any three of the five keys that were created.

    You must run the `vault operator unseal` command for each of the three keys:

    a. Unseal vault by using the first key:

    ```
    [root@mcctb ~]# vault operator unseal
    Unseal Key (will be hidden):
    Key                Value
    ---                -----
    Seal Type          shamir
    Initialized        true
    Sealed             true
    Total Shares       5
    Threshold          3
    Unseal Progress    1/3
    Unseal Nonce       <unseal_key_1_id>
    Version            1.12.2
    Build Date         2022-11-23T12:53:46Z
    Storage Type       file
    HA Enabled         false
    ```

    b. Unseal vault by using the second key:

    ```
    [root@mcctb ~]# vault operator unseal
    Unseal Key (will be hidden):
    Key                Value
    ---                -----
    Seal Type          shamir
    Initialized        true
    Sealed             true
    Total Shares       5
    Threshold          3
    Unseal Progress    2/3
    Unseal Nonce       <unseal_key_2_id>
    Version            1.12.2
    Build Date         2022-11-23T12:53:46Z
    Storage Type       file
    HA Enabled         false
    ```

    c. Unseal vault by using the third key:

```
[root@mcctb ~]# vault operator unseal
Unseal Key (will be hidden):
2022-12-15T15:15:00.980+0530 [INFO]  core.cluster-listener.tcp:
starting listener: listener_address=127.0.0.1:8201
2022-12-15T15:15:00.980+0530 [INFO]  core.cluster-listener: serving
cluster requests: cluster_listen_address=127.0.0.1:8201
2022-12-15T15:15:00.981+0530 [INFO]  core: post-unseal setup starting
2022-12-15T15:15:00.981+0530 [INFO]  core: loaded wrapping token key
2022-12-15T15:15:00.982+0530 [INFO]  core: successfully setup plugin
catalog: plugin-directory=""
2022-12-15T15:15:00.983+0530 [INFO]  core: successfully mounted
backend: type=system version="" path=sys/
2022-12-15T15:15:00.984+0530 [INFO]  core: successfully mounted
backend: type=identity version="" path=identity/
2022-12-15T15:15:00.984+0530 [INFO]  core: successfully mounted
backend: type=cubbyhole version="" path=cubbyhole/
2022-12-15T15:15:00.986+0530 [INFO]  core: successfully enabled
credential backend: type=token version="" path=token/ namespace="ID:
root. Path: "
2022-12-15T15:15:00.986+0530 [INFO]  rollback: starting rollback
manager
2022-12-15T15:15:00.987+0530 [INFO]  core: restoring leases
2022-12-15T15:15:00.987+0530 [INFO]  expiration: lease restore
complete
2022-12-15T15:15:00.987+0530 [INFO]  identity: entities restored
2022-12-15T15:15:00.987+0530 [INFO]  identity: groups restored
2022-12-15T15:15:00.988+0530 [INFO]  core: usage gauge collection is
disabled
2022-12-15T15:15:00.989+0530 [INFO]  core: post-unseal setup complete
2022-12-15T15:15:00.989+0530 [INFO]  core: vault is unsealed
Key                Value
---                -----
Seal Type          shamir
Initialized        true
Sealed             false
Total Shares       5
Threshold          3
Version            1.12.2
Build Date         2022-11-23T12:53:46Z
Storage Type       file
Cluster Name       vault-cluster
Cluster ID         <cluster_id>
HA Enabled         false
```

12. Verify that the Vault sealed status is false.

```
[root@mcctb ~]# vault status
Key                Value
---                -----
Seal Type          shamir
Initialized        true
Sealed             false
Total Shares       5
Threshold          3
Version            1.12.2
Build Date         2022-11-23T12:53:46Z
Storage Type       file
Cluster Name       vault-cluster
Cluster ID         <cluster_id>
HA Enabled         false
```

13. Configure the Vault service to start on boot.

    a. Run the following command: `cd /etc/systemd/system`

    ```
    [root@mcctb ~]#  cd /etc/systemd/system
    ```

    b. At the `[root@mcctb system]` prompt, copy and run the following command to create the Vault service file.

    ```
    # cat > vault.service << EOF
    [Unit]
    Description=Vault Service
    After=mariadb.service

    [Service]
    Type=forking
    ExecStart=/usr/bin/vault server -config /root/config.hcl &
    Restart=on-failure

    [Install]
    WantedBy=multi-user.target
    EOF
    ```

    c. Run the following command: `systemctl daemon-reload`

    ```
    [root@mcctb system]#  systemctl daemon-reload
    ```

    d. Run the following command: `systemctl enable vault.service`

```
[root@mcctb system]#  systemctl enable vault.service
Created symlink /etc/systemd/system/multi-
user.target.wants/vault.service → /etc/systemd/system/vault.service.
```

You are prompted to use this feature during the installation of MetroCluster Tiebreaker. If you want to change the method to unseal Vault, then you need to uninstall and reinstall the MetroCluster Tiebreaker software.

## Installing MySQL Server 5.5.30 or later and 5.6.x versions on Red Hat Enterprise Linux 7 or CentOS 7

You must install MySQL Server 5.5.30 or later and 5.6.x version on your host system before installing or upgrading the Tiebreaker software.

**Steps**

1. Log in as a root user or a sudo user that can change to advanced privilege mode.

   ```
   login as: root
   root@mcctb's password:
   Last login: Fri Jan  8 21:33:00 2016 from host.domain.com
   ```

2. Add the MySQL repository to your host system:

   ```
   [root@mcctb ~]# yum localinstall https://dev.mysql.com/get/mysql57-community-
   release-el6-11.noarch.rpm
   ```

```
Loaded plugins: product-id, refresh-packagekit, security, subscription-
manager
Setting up Local Package Process
Examining /var/tmp/yum-root-LLUw0r/mysql-community-release-el6-
5.noarch.rpm: mysql-community-release-el6-5.noarch
Marking /var/tmp/yum-root-LLUw0r/mysql-community-release-el6-
5.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package mysql-community-release.noarch 0:el6-5 will be installed
--> Finished Dependency Resolution
Dependencies Resolved
========================================================================
========
Package               Arch   Version
                                    Repository
Size
========================================================================
========
Installing:
mysql-community-release
                       noarch el6-5 /mysql-community-release-el6-
5.noarch 4.3 k
Transaction Summary
========================================================================
========
Install       1 Package(s)
Total size: 4.3 k
Installed size: 4.3 k
Is this ok [y/N]: y
Downloading Packages:
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : mysql-community-release-el6-5.noarch
1/1
  Verifying  : mysql-community-release-el6-5.noarch
1/1
Installed:
  mysql-community-release.noarch 0:el6-5
Complete!
```

3. Disable the MySQL 57 repository:

```
[root@mcctb ~]# yum-config-manager --disable mysql57-community
```

4. Enable the MySQL 56 repository:

```
[root@mcctb ~]# yum-config-manager --enable mysql56-community
```

5. Enable the repository:

```
[root@mcctb ~]# yum repolist enabled | grep "mysql.-community."
```

```
mysql-connectors-community           MySQL Connectors Community
21
mysql-tools-community                MySQL Tools Community
35
mysql56-community                    MySQL 5.6 Community Server
231
```

6. Install the MySQL Community server:

```
[root@mcctb ~]# yum install mysql-community-server
```

```
Loaded plugins: product-id, refresh-packagekit, security, subscription-
manager
This system is not registered to Red Hat Subscription Management. You
can use subscription-manager
to register.
Setting up Install Process
Resolving Dependencies
--> Running transaction check
 .....Output truncated.....
---> Package mysql-community-libs-compat.x86_64 0:5.6.29-2.el6 will be
obsoleting
--> Finished Dependency Resolution
Dependencies Resolved
========================================================================
======
Package                          Arch   Version       Repository
Size
========================================================================
======
Installing:
 mysql-community-client         x86_64  5.6.29-2.el6  mysql56-community
18  M
     replacing  mysql.x86_64 5.1.71-1.el6
 mysql-community-libs           x86_64  5.6.29-2.el6  mysql56-community
1.9 M
```

```
     replacing  mysql-libs.x86_64 5.1.71-1.el6
 mysql-community-libs-compat    x86_64  5.6.29-2.el6  mysql56-community
1.6 M
     replacing  mysql-libs.x86_64 5.1.71-1.el6
 mysql-community-server         x86_64  5.6.29-2.el6  mysql56-community
53  M
     replacing  mysql-server.x86_64 5.1.71-1.el6
Installing for dependencies:
mysql-community-common          x86_64  5.6.29-2.el6  mysql56-community
308 k

Transaction Summary
========================================================================
=======
Install       5 Package(s)
Total download size: 74 M
Is this ok [y/N]: y
Downloading Packages:
(1/5): mysql-community-client-5.6.29-2.el6.x86_64.rpm       |  18 MB
00:28
(2/5): mysql-community-common-5.6.29-2.el6.x86_64.rpm       | 308 kB
00:01
(3/5): mysql-community-libs-5.6.29-2.el6.x86_64.rpm         | 1.9 MB
00:05
(4/5): mysql-community-libs-compat-5.6.29-2.el6.x86_64.rpm  | 1.6 MB
00:05
(5/5): mysql-community-server-5.6.29-2.el6.x86_64.rpm       |  53 MB
03:42
 -----------------------------------------------------------------------
--------
Total                                              289 kB/s |  74 MB
04:24
warning: rpmts_HdrFromFdno: Header V3 DSA/SHA1 Signature, key ID
<key_id> NOKEY
Retrieving key from file:/etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
Importing GPG key 0x5072E1F5:
 Userid : MySQL Release Engineering <mysql-build@oss.oracle.com>
Package: mysql-community-release-el6-5.noarch
         (@/mysql-community-release-el6-5.noarch)
 From   : file:/etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
Is this ok [y/N]: y
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : mysql-community-common-5.6.29-2.el6.x86_64
```

```
....Output truncated....
1.el6.x86_64
7/8
  Verifying  : mysql-5.1.71-1.el6.x86_64
8/8
Installed:
  mysql-community-client.x86_64 0:5.6.29-2.el6
  mysql-community-libs.x86_64 0:5.6.29-2.el6
  mysql-community-libs-compat.x86_64 0:5.6.29-2.el6
  mysql-community-server.x86_64 0:5.6.29-2.el6

Dependency Installed:
  mysql-community-common.x86_64 0:5.6.29-2.el6

Replaced:
  mysql.x86_64 0:5.1.71-1.el6 mysql-libs.x86_64 0:5.1.71-1.el6
  mysql-server.x86_64 0:5.1.71-1.el6
Complete!
```

7. Start MySQL server:

```
[root@mcctb ~]# service mysqld start
```

```
Initializing MySQL database:  2016-04-05 19:44:38 0 [Warning] TIMESTAMP
with implicit DEFAULT value is deprecated. Please use
--explicit_defaults_for_timestamp server option (see documentation
for more details).
2016-04-05 19:44:38 0 [Note] /usr/sbin/mysqld (mysqld 5.6.29)
        starting as process 2487 ...
2016-04-05 19:44:38 2487 [Note] InnoDB: Using atomics to ref count
        buffer pool pages
2016-04-05 19:44:38 2487 [Note] InnoDB: The InnoDB memory heap is
disabled
....Output truncated....
2016-04-05 19:44:42 2509 [Note] InnoDB: Shutdown completed; log sequence
       number 1625987

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER!
To do so, start the server, then issue the following commands:

  /usr/bin/mysqladmin -u root password 'new-password'
  /usr/bin/mysqladmin -u root -h mcctb password 'new-password'

Alternatively, you can run:
  /usr/bin/mysql_secure_installation

which will also give you the option of removing the test
databases and anonymous user created by default.  This is
strongly recommended for production servers.
.....Output truncated.....
WARNING: Default config file /etc/my.cnf exists on the system
This file will be read by default by the MySQL server
If you do not want to use this, either remove it, or use the
--defaults-file argument to mysqld_safe when starting the server

                                                           [  OK  ]
Starting mysqld:                                           [  OK  ]
```

8. Confirm that MySQL server is running:

```
[root@mcctb ~]# service mysqld status
```

```
mysqld (pid  2739) is running...
```

9. Configure security and password settings:

```
[root@mcctb ~]# mysql_secure_installation
```

```
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
       SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

 In order to log into MySQL to secure it, we'll need the current
 password for the root user.  If you've just installed MySQL, and
 you haven't set the root password yet, the password will be blank,
 so you should just press enter here.

 Enter current password for root (enter for none):  <== on default
install
                                                         hit enter here
 OK, successfully used password, moving on...

 Setting the root password ensures that nobody can log into the MySQL
 root user without the proper authorization.

 Set root password? [Y/n] y
 New password:
 Re-enter new password:
 Password updated successfully!
 Reloading privilege tables..
  ... Success!

 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? [Y/n] 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? [Y/n] 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? [Y/n] y
  - Dropping test database...
 ERROR 1008 (HY000) at line 1: Can't drop database 'test';
```

```
database doesn't exist
 ... Failed!  Not critical, keep moving...
 - 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!

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

Thanks for using MySQL!

Cleaning up...
```

10. Verify that the MySQL login is working:

```
[root@mcctb ~]# mysql -u root -p
```

```
Enter password: <configured_password>
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 5.6.29 MySQL Community Server (GPL)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights
reserved.

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>
```

If the MySQL login is working, the output will end at the `mysql>` prompt.

### Enabling the MySQL autostart setting

You should verify that the autostart feature is turned on for the MySQL daemon. Turning on the MySQL daemon automatically restarts MySQL if the system on which the MetroCluster Tiebreaker software resides reboots. If the MySQL daemon is not running, the Tiebreaker software continues running, but it cannot be restarted and configuration changes cannot be made.

**Step**

1. Verify that MySQL is enabled to autostart when booted:

   ```
   [root@mcctb ~]# systemctl list-unit-files mysqld.service
   ```

   ```
   UNIT FILE          State
   ------------------ ----------
   mysqld.service     enabled
   ```

   If MySQL is not enabled to autostart when booted, see the MySQL documentation to enable the autostart feature for your installation.

## Installing MariaDB server on Red Hat Enterprise Linux 8

You must install MariaDB server on your host system before installing or upgrading the Tiebreaker software.

**Before you begin**

Your host system must be running on Red Hat Enterprise Linux (RHEL) 8.

**Steps**

1. Log in as a `root` user or a user that can sudo to advanced privilege mode.

```
login as: root
root@mcctb's password:
Last login: Fri Jan  8 21:33:00 2017 from host.domain.com
```

1. Install the MariaDB server:

   ```
   [root@mcctb ~]# yum install mariadb-server.x86_64
   ```

   ```
   [root@mcctb ~]# yum install mariadb-server.x86_64
   Loaded plugins: fastestmirror, langpacks
    ...
    ...

   ========================================================================
   ===
    Package                      Arch   Version         Repository
   Size
   ========================================================================
   ===
   Installing:
   mariadb-server               x86_64   1:5.5.56-2.el7   base
   11 M
   Installing for dependencies:
   ```

```
Transaction Summary
========================================================================
===
Install  1 Package  (+8 Dependent packages)
Upgrade             ( 1 Dependent package)

Total download size: 22 M
Is this ok [y/d/N]: y

Downloading packages:
No Presto metadata available for base warning:
/var/cache/yum/x86_64/7/base/packages/mariadb-libs-5.5.56-
2.el7.x86_64.rpm:
Header V3 RSA/SHA256 Signature,
key ID f4a80eb5: NOKEY] 1.4 MB/s | 3.3 MB  00:00:13 ETA
Public key for mariadb-libs-5.5.56-2.el7.x86_64.rpm is not installed
(1/10): mariadb-libs-5.5.56-2.el7.x86_64.rpm  | 757 kB  00:00:01
..
..
(10/10): perl-Net-Daemon-0.48-5.el7.noarch.rpm|  51 kB  00:00:01
------------------------------------------------------------------------
-----------------
Installed:
  mariadb-server.x86_64 1:5.5.56-2.el7

Dependency Installed:
mariadb.x86_64 1:5.5.56-2.el7
perl-Compress-Raw-Bzip2.x86_64 0:2.061-3.el7
perl-Compress-Raw-Zlib.x86_64 1:2.061-4.el7
perl-DBD-MySQL.x86_64 0:4.023-5.el7
perl-DBI.x86_64 0:1.627-4.el7
perl-IO-Compress.noarch 0:2.061-2.el7
perl-Net-Daemon.noarch 0:0.48-5.el7
perl-PlRPC.noarch 0:0.2020-14.el7

Dependency Updated:
  mariadb-libs.x86_64 1:5.5.56-2.el7
Complete!
```

2. Start MariaDB server:

```
[root@mcctb ~]# systemctl start mariadb
```

3. Verify that the MariaDB server has started:

```
[root@mcctb ~]# systemctl status mariadb
```

```
[root@mcctb ~]# systemctl status mariadb
mariadb.service - MariaDB database server
...
Nov 08 21:28:59 mcctb systemd[1]: Starting MariaDB database server...
...
Nov 08 21:29:01 mcctb systemd[1]: Started MariaDB database server.
```

4. Configure the security and password settings:

   When you are prompted for the root password, leave it empty and press enter to continue to configure the security and password settings.

```
[root@mcctb ~]# mysql_secure_installation
```

```
root@localhost systemd]# mysql_secure_installation

 NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user.  If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

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

Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

Set root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
 ... Success!


By default, a MariaDB installation has an anonymous user, allowing
anyone
to log into MariaDB 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? [Y/n] 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? [Y/n] y
 ... Success!

By default, MariaDB 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? [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]

 ... Success!

Cleaning up...

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

Thanks for using MariaDB!
```

### Enabling the autostart setting for the MariaDB server

You should verify that the autostart feature is turned on for the MariaDB server. If you do not enable the autostart feature, and the system on which the MetroCluster Tiebreaker software resides has to reboot, then the Tiebreaker software continues running, but the MariaDB service cannot be restarted and configuration changes cannot be made.

#### Steps

1. Enable the autostart service:

   ```
   [root@mcctb ~]# systemctl enable mariadb.service
   ```

2. Verify that MariaDB is enabled to autostart when booted:

   ```
   [root@mcctb ~]# systemctl list-unit-files mariadb.service
   ```

```
UNIT FILE          State
------------------ ----------
mariadb.service    enabled
```

# Installing or upgrading the software package

You must install or upgrade the MetroCluster Tiebreaker software on your local computer to monitor MetroCluster configurations.

### About this task

- Your storage system must be running ONTAP 8.3.x or later.
- You must have installed OpenJDK by using the `yum install java-x.x.x-openjdk` command. For systems running Tiebreaker 1.4 and earlier, run JRE 8. For systems running Tiebreaker 1.5 and later, run OpenJDK 17, 18, or 19. The outputs in the example show JRE 1.8.0. (JRE 8).
- You can install MetroCluster Tiebreaker as a non-root user with sufficient administrative privileges to perform the Tiebreaker installation, create tables and users, and set the user password.

### Steps

1. Download the MetroCluster Tiebreaker software. This example uses version 1.5.

   NetApp Support Site

2. Download the `MetroCluster_Tiebreaker_RPM_GPG` key:

   NetApp Support Site

3. Log in to the host as the root user.
4. Create a non-root user and the `mcctbgrp` group.

   a. Create a non-root user and set the password.

      The following example commands create a non-root user named `mcctbuser1`:

      ```
      [root@mcctb ~]# useradd mcctbuser1
      [root@mcctb ~]# passwd mcctbuser1
      Changing password for user mcctbuser1.
      New password:
      Retype new password:
      passwd: all authentication tokens updated successfully.
      ```

   b. Create a group named `mcctbgrp`:

      ```
      [root@mcctb ~~]# groupadd mcctbgrp
      ```

   c. Add the non-root user you created to the `mcctbgrp` group.

The following command adds `mcctbuser1` to the `mcctbgrp` group:

```
[root@mcctb ~]# usermod -a -G mcctbgrp mcctbuser1
```

5. Verify the RPM file.

   Run the following substeps from the directory containing the RPM key.

   a. Download and import the RPM key file:

   ```
   [root@mcctb ~]# rpm --import MetroCluster_Tiebreaker_RPM_GPG.key
   ```

   b. Verify the that the correct key was imported by checking the fingerprint.

   The following example shows a correct key fingerprint:

   ```
   root@mcctb:~/signing/mcctb-rpms# gpg --show-keys --with-fingerprint
   MetroCluster_Tiebreaker_RPM_GPG.key
   pub   rsa3072 2022-11-17 [SCEA] [expires: 2025-11-16]
         65AC 1562 E28A 1497 7BBD  7251 2855 EB02 3E77 FAE5
   uid                      MCCTB-RPM (mcctb RPM production signing)
   <mcctb-rpm@netapp.com>
   ```

   c. Verify the signature: `rpm --checksig NetApp-MetroCluster-Tiebreaker-Software-1.5-1.x86_64.rpm`

   ```
   NetApp-MetroCluster-Tiebreaker-Software-1.5-1.x86_64.rpm: digests OK
   ```

   > You must only proceed with installation after you have successfully verified the signature.

6. Install or upgrade the Tiebreaker software:

   > You can only upgrade to Tiebreaker version 1.5 when you are upgrading from Tiebreaker version 1.4. Upgrading from earlier versions to Tiebreaker 1.5 is not supported.

   Select the correct procedure from below depending on whether you’re performing a new installation or upgrading an existing installation.

**Perform a new installation**

a. Retrieve and record the absolute path for Java:

```
[root@mcctb ~]# readlink -f /usr/bin/java
/usr/lib/jvm/java-19-openjdk-19.0.0.0.36-
2.rolling.el8.x86_64/bin/java
```

b. Run the following command: `rpm -ivh NetApp-MetroCluster-Tiebreaker-Software-1.5-1.x86_64.rpm`

The system displays the following output for a successful installation:

> ⓘ When prompted during the installation, provide the non-root user that you previously created and assigned to the `mcctbgrp` group.

```
Verifying...
################################# [100%]
Preparing...
################################# [100%]
Updating / installing...
   1:NetApp-MetroCluster-Tiebreaker-
So################################# [100%]
Enter the absolute path for Java : /usr/lib/jvm/java-19-openjdk-
19.0.0.0.36-2.rolling.el8.x86_64/bin/java
Verifying if Java exists...
Found Java. Proceeding with the installation.
Enter host user account to use for the installation:
mcctbuser1
User account mcctbuser1 found. Proceeding with the installation
Enter database user name:
root
Please enter database password for root
Enter password:
Sealed          false
Do you wish to auto unseal vault(y/n)?y
Enter the key1:
Enter the key2:
Enter the key3:
Success! Uploaded policy: mcctb-policy
Error enabling approle auth: Error making API request.
URL: POST http://127.0.0.1:8200/v1/sys/auth/approle
Code: 400. Errors:
* path is already in use at approle/
Success! Enabled the kv secrets engine at: mcctb/
Success! Data written to: auth/approle/role/mcctb-app
Password updated successfully in the vault.
Synchronizing state of netapp-metrocluster-tiebreaker-
software.service with SysV service script with
/usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable netapp-
metrocluster-tiebreaker-software
Created symlink /etc/systemd/system/multi-
user.target.wants/netapp-metrocluster-tiebreaker-software.service
→ /etc/systemd/system/netapp-metrocluster-tiebreaker-
software.service.
Attempting to start NetApp MetroCluster Tiebreaker software
services
Started NetApp MetroCluster Tiebreaker software services
Successfully installed NetApp MetroCluster Tiebreaker software
version 1.5.
```

**Upgrading an existing installation**

a. Verify that a supported version of OpenJDK is installed and is the current Java version located on the host.

> For upgrades to Tiebreaker 1.5, you must install either OpenJDK version 17, 18, or 19.

```
[root@mcctb ~]# readlink -f /usr/bin/java
/usr/lib/jvm/java-19-openjdk-19.0.0.0.36-
2.rolling.el8.x86_64/bin/java
```

b. Verify the Vault service is unsealed and running: `vault status`

```
[root@mcctb ~]# vault status
Key                Value
---                -----
Seal Type          shamir
Initialized        true
Sealed             false
Total Shares       5
Threshold          3
Version            1.12.2
Build Date         2022-11-23T12:53:46Z
Storage Type       file
Cluster Name       vault
Cluster ID         <cluster_id>
HA Enabled         false
```

c. Upgrade the Tiebreaker software.

```
[root@mcctb ~]# rpm -Uvh NetApp-MetroCluster-Tiebreaker-Software-
1.5-1.x86_64.rpm
```

The system displays the following output for a successful upgrade:

```
Verifying...
################################# [100%]
Preparing...
################################# [100%]
Updating / installing...
   1:NetApp-MetroCluster-Tiebreaker-
So################################# [ 50%]

Enter the absolute path for Java : /usr/lib/jvm/java-19-openjdk-
19.0.0.0.36-2.rolling.el8.x86_64/bin/java
Verifying if Java exists...
Found Java. Proceeding with the installation.
Enter host user account to use for the installation:
mcctbuser1
User account mcctbuser1 found. Proceeding with the installation
Sealed          false
Do you wish to auto unseal vault(y/n)?y
Enter the key1:
Enter the key2:
Enter the key3:
Success! Uploaded policy: mcctb-policy
Error enabling approle auth: Error making API request.
URL: POST http://127.0.0.1:8200/v1/sys/auth/approle
Code: 400. Errors:
* path is already in use at approle/
Success! Enabled the kv secrets engine at: mcctb/
Success! Data written to: auth/approle/role/mcctb-app
Enter database user name : root
Please enter database password for root
Enter password:
Password updated successfully in the database.
Password updated successfully in the vault.
Synchronizing state of netapp-metrocluster-tiebreaker-
software.service with SysV service script with
/usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable netapp-
metrocluster-tiebreaker-software
Attempting to start NetApp MetroCluster Tiebreaker software
services
Started NetApp MetroCluster Tiebreaker software services
Successfully upgraded NetApp MetroCluster Tiebreaker software to
version 1.5.
Cleaning up / removing...
   2:NetApp-MetroCluster-Tiebreaker-
So################################# [100%]
```

ⓘ If you enter the wrong MySQL root password, the Tiebreaker software indicates that it was installed successfully, but displays "Access denied" messages. To resolve the issue, you must uninstall the Tiebreaker software by using the `rpm -e` command, and then reinstall the software by using the correct MySQL root password.

7. Check the Tiebreaker connectivity to the MetroCluster software by opening an SSH connection from the Tiebreaker host to each of the node management LIFs and cluster management LIFs.

**Related information**

NetApp Support