安装 Tiebreaker 软件

ONTAP MetroCluster

NetApp
April 25, 2024

目录

安装 Tiebreaker 软件

Tieb破碎 机安装工作流

Tiebreaker 软件可为集群存储环境提供监控功能。此外，它还会在发生节点连接问题和站点灾难时发送 SNMP 通知。

关于此工作流

您可以使用此工作流安装或升级Tieb破碎 机软件。

1. "准备安装Tieb破碎 机软件"

在安装和配置Tieb破碎 机软件之前、请确认您的系统满足特定要求。

2. "确保安装安全"

对于运行MetroCluster Tiebreaker 1.5及更高版本的配置、您可以保护和强化主机操作系统和数据库。

3. "安装Tieb破碎 机软件包"

重新安装或升级Tieberr软件。您遵循的安装操作步骤取决于您要安装的Tieb破碎 机版本。

准备安装Tieb破碎 机软件

在安装和配置Tieb破碎 机软件之前、您应验证您的系统是否满足特定要求。

软件要求

您必须满足以下软件要求、具体取决于您要安装的Tieb破碎 机版本。

ONTAP Tieb破碎 机版本	支持的ONTAP版本	支持的 Linux 版本	Java/MariaDB要求
1.6.	ONTAP 9.12.1及更高版本	请参见 "操作系统支持列表" 了解详细信息。	无依赖关系与安装捆绑在一起。
1.5	ONTAP 9.8 到ONTAP 9.14.1	• Red Hat Enterprise Linux 8.1至8.7	对于Red Hat Enterprise Linux 8.1至8.7： • MariaDB 10.x (使用使用"yum install MariaDB-server.x86_64 "安装的默认版本) • OpenJDK 17、18或19

1.4.	ONTAP 9.1 到ONTAP 9.9.1	• Red Hat Enterprise Linux 8.1至8.7 • Red Hat Enterprise Linux 7至7.9 • CentOS 7到7.9 (64位)	使用CentOS： • MariaDB 5.5.52.x/MySQL Server 5.6x • 4 GB RAM • 打开JRE 8 对于Red Hat Enterprise Linux 8.1至8.7： • MariaDB 10.x (使用使用"yum install MariaDB-server.x86_64 "安装的默认版本) • JRE 8.

其他要求

您必须注意以下附加要求：

- Tiebreaker 软件安装在第三个站点上，该软件可以区分交换机间链路（ ISL ）故障（站点间链路关闭时）和站点故障。您的主机系统必须满足特定要求、然后才能安装或升级Tieb破碎 机软件以监控MetroCluster配置。
- 要安装MetroCluster Tiebreaker软件和相关软件包、您必须具有"root"权限。
- 每个MetroCluster 配置只能使用一个MetroCluster Tiebreaker监控器、以避免与多个Tiebreaker监控器发生任何冲突。
- 在为Tieb破碎 机软件选择网络时间协议(Network Time Protocol、NTP)源时、必须使用本地NTP源。Tieb破碎 机软件不应使用与Tieb破碎 机软件监控的MetroCluster站点相同的源。
- 磁盘容量： 8 GB
- 防火墙：
  - 直接访问以设置 AutoSupport 消息
  - SSH （端口 22/TCP ）， HTTPS （端口 443/TCP ）和 ping （ ICMP ）

确保Tieb破碎 机主机和数据库安装的安全

对于运行MetroCluster Tiebreaker 1.5及更高版本的配置、您可以保护和强化主机操作系统和数据库。

保护主机的安全

以下准则介绍了如何保护安装Tiebreaker软件的主机的安全。

用户管理建议

- 限制"root"用户的访问。
  - 您可以使用能够提升到root访问权限的用户来安装和管理Tiebreaker软件。

- 您可以使用无法提升为root访问权限的用户来管理Tiebreaker软件。
- 安装期间、必须创建一个名为"mcctbgrp"的组。主机root用户和安装期间创建的用户都必须是成员。只有此组的成员才能完全管理Tiebreaker软件。

> 非此组成员用户无法访问Tiebreaker软件或命令行界面。您可以在主机上创建其他用户并使其成为组的成员。这些附加成员无法完全管理Tieber-软件。他们具有只读访问权限、无法添加、更改或删除显示器。

- 请勿以root用户身份运行Tiebreaker。使用专用的无特权服务帐户运行Tiebreaker。

- 更改/etc/snmp/snmpd.conf文件中的默认社区字符串。
- 允许最小写入权限。无权限的Tiebreaker服务帐户不应有权覆盖其可执行二进制文件或任何配置文件。只有本地Tiebreaker存储(例如、集成后端存储)的目录和文件或审核日志才应由Tiebreaker用户写入。
- 不允许匿名用户。
  - 将AllowTcpForwarding设置为"no"或使用match指令限制匿名用户。

相关信息

- "Red Hat Enterprise Linux 8产品文档"
- "Red Hat Enterprise Linux 9产品文档"

基线主机安全建议

- 使用磁盘加密
  - 您可以启用磁盘加密。这可以是由Hostos (软件)或SVM主机提供的FullDiskEncryption (硬件)或加密。
- 禁用允许传入连接的未使用服务。您可以禁用任何未使用的服务。Tiebreaker软件不需要为传入连接提供服务、因为Tiebreaker安装中的所有连接都是传出的。默认情况下可能启用且可以禁用的服务包括：
  - HTTP/HTTPS服务器
  - FTP服务器
  - Telnet、RSH、rlogin
  - NFS、CIFS和其他协议访问
  - RDP (RemoteDesktopProtocol、X11 Server、VNC或其他远程"桌面"服务提供程序。

> 要远程管理主机、您必须保持串行控制台访问(如果支持)或至少启用一个协议。如果禁用所有协议、则需要对主机进行物理访问以进行管理。

- 使用FIPS保护主机安全
  - 您可以在FIPS兼容模式下安装主机操作系统、然后安装Tiebreaker。

> OpenJDK 19会在启动时检查主机是否安装在FIPS模式下。无需手动更改。

  - 如果您保护主机安全、则必须确保主机能够在没有用户干预的情况下启动。如果需要用户干预、则在主机意外重新启动时、Tieb破碎 机功能可能不可用。如果发生这种情况、Tieb破碎 机功能仅在手动干预后以及主机完全启动后可用。

- 禁用Shell命令历史记录。
- 请经常升级。Tiebreaker是一款主动开发的解决方案、经常更新对于整合安全修复以及对默认设置(如密钥长度或密码套件)进行的任何更改非常重要。
- 订阅HashiCorp公告邮件列表以接收新版本的公告、并访问Tiebreaker ChangeLog以了解有关新版本最新更新的详细信息。
- 使用正确的文件权限。在启动Tiebreaker软件之前、请始终确保对文件应用适当的权限、尤其是包含敏感信息的文件。
- 多因素身份验证(MultiFactor Authentication、MFA)要求管理员使用多个用户名和密码来识别自己、从而增强了组织的安全性。用户名和密码虽然重要、但容易受到暴力攻击、并可能被第三方窃取。
  - Red Hat Enterprise Linux 8提供了MFA、它要求用户提供多条信息、以便成功向帐户或Linux主机进行身份验证。追加信息 可能是通过SMS或Google Authenticator、Twilio Authy或FreeOTP等应用程序的凭据一次性发送到您的手机的密码。

相关信息

- "Red Hat Enterprise Linux 8产品文档"
- "Red Hat Enterprise Linux 9产品文档"

保护数据库安装的安全

以下准则说明了如何保护和强化MariaDB 10.x数据库安装。

- 限制"root"用户的访问。
  - Tiebreaker使用专用帐户。用于存储(配置)数据的帐户和表是在安装Tiebreaker期间创建的。只需要在安装期间提升对数据库的访问权限。
- 在安装期间、需要以下访问和权限：
  - 创建数据库和表的功能
  - 创建全局选项的功能
  - 创建数据库用户并设置密码的功能
  - 能够将数据库用户与数据库和表关联并分配访问权限

> 在Tiebreaker安装期间指定的用户帐户必须具有所有这些特权。不支持对不同任务使用多个用户帐户。

- 对数据库使用加密
  - 支持空闲数据加密。 "详细了解空闲数据加密"
  - 传输中的数据未加密。传输中的数据使用本地"SOCs"文件连接。
  - MariaDB的FIPS兼容性—您不需要在数据库上启用FIPS兼容性。在FIPS兼容模式下安装主机即可。

    "了解MySQL企业级透明数据加密(TDE)"

> 在安装TiebrAKER软件之前、必须启用加密设置。

相关信息

- 数据库用户管理

  "访问控制和帐户管理"

- 保护数据库的安全

  "使MySQL安全防范攻击者攻击"

  "保护MariaDB的安全"

- 保护存储安装的安全

  "生产强化"

安装Tieb破碎 机软件包

选择您的安装操作步骤

您遵循的Tieb破碎 机安装操作步骤取决于您正在安装的Tieb破碎 机版本。

Tieb破碎 机版本	转至 …
Tieb破碎 机1.6	"安装Tieb破碎 锤1.6"
Tieb破碎 机1.5	"安装Tieb破碎 锤1.5"
Tieb破碎 机1.4	"安装Tieb破碎 锤1.4"

安装Tieb破碎 锤1.6

在主机Linux操作系统上全新安装或升级到Tieb破碎 机1.6以监控MetroCluster配置。

关于此任务

- 存储系统必须运行ONTAP 9.12.1或更高版本。
- 您可以使用具有足够管理权限的非root用户身份安装MetroCluster Tieb破碎 镐、以便执行Tieb破碎 镐安装、创建表和用户以及设置用户密码。

步骤

1. 下载MetroCluster Tieb破碎 机1.6软件。

   "MetroCluster Tieb破碎 机(下载)—NetApp 支持站点"

2. 以 root 用户身份登录到主机。
3. 如果要执行升级、请验证所运行的Tieb破碎 机版本：

   以下示例显示了Tieb破碎 机1.5。

```
[root@mcctb ~] # netapp-metrocluster-tiebreaker-software-cli
NetApp MetroCluster Tiebreaker :> version show
NetApp MetroCluster Tiebreaker 1.5: Sun Mar 13 09:59:02 IST 2022
NetApp MetroCluster Tiebreaker :> exit
```

4. 安装或升级Tieb破碎 机软件。

安装Tieb破碎 锤1.6

按照以下步骤全新安装Tieb破碎 机1.6。

步骤

a. 在运行以下命令 `[root@mcctb ~] #` 提示您开始安装：

```
sh MetroClusterTiebreakerInstall-1.6
```

成功安装时，系统将显示以下输出：

```
Extracting the MetroCluster Tiebreaker installation/upgrade
archive
Install digest hash is Ok
Performing the MetroCluster Tiebreaker code signature check
Install code signature is Ok
Enter unix user account to use for the installation:
mcctbadminuser
Unix user account "mcctbadminuser" doesn't exist. Do you wish to
create "mcctbadminuser" user account? [Y/N]: y
useradd: warning: the home directory already exists.
Not copying any file from skel directory into it.
Creating mailbox file: File exists
Unix account "mcctbadminuser" created.
Changing password for user mcctbadminuser.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
MetroCluster Tiebreaker requires unix user account
"mcctbadminuser" to be added to the group "mcctbgrp" for admin
access.
Do you wish to add ? [Y/N]: y
Unix user account "mcctbadminuser" added to "mcctbgrp".
Do you wish to generate your own public-private key pair for
encrypting audit log? [Y/N]: y
Generating public-private key pair...
Configuring Vault...
Starting vault server...
==> Vault server configuration:

             Api Address: <api_address>
                     Cgo: disabled
         Cluster Address: <cluster_address>
   Environment Variables: BASH_FUNC_which%%,
DBUS_SESSION_BUS_ADDRESS, GODEBUG, HISTCONTROL, HISTSIZE, HOME,
```

```
HOSTNAME, HOST_ACCOUNT, LANG, LESSOPEN, LOGNAME, LS_COLORS, MAIL,
PATH, PWD, SHELL, SHLVL, SSH_CLIENT, SSH_CONNECTION, SSH_TTY,
STAF_TEMP_DIR, TERM, USER, VAULT_ADDR, VAULT_TOKEN,
XDG_RUNTIME_DIR, XDG_SESSION_ID, _, vault_Addr, which_declare
              Go Version: go1.20.5
              Listener 1: tcp (addr: "0.0.0.0:8200", cluster
address: "0.0.0.0:8201", max_request_duration: "1m30s",
max_request_size: "33554432", tls: "enabled")
               Log Level:
                   Mlock: supported: true, enabled: true
           Recovery Mode: false
                 Storage: file
                 Version: Vault v1.14.0, built 2023-06-
19T11:40:23Z
             Version Sha:
13a649f860186dffe3f3a4459814d87191efc321

==> Vault server started! Log data will stream in below:

2023-11-23T15:14:28.532+0530 [INFO]  proxy environment:
http_proxy="" https_proxy="" no_proxy=""
2023-11-23T15:14:28.577+0530 [INFO]  core: Initializing version
history cache for core
2023-11-23T15:14:38.552+0530 [INFO]  core: security barrier not
initialized
2023-11-23T15:14:38.552+0530 [INFO]  core: seal configuration
missing, not initialized
2023-11-23T15:14:38.554+0530 [INFO]  core: security barrier not
initialized
2023-11-23T15:14:38.555+0530 [INFO]  core: security barrier
initialized: stored=1 shares=5 threshold=3
2023-11-23T15:14:38.556+0530 [INFO]  core: post-unseal setup
starting
2023-11-23T15:14:38.577+0530 [INFO]  core: loaded wrapping token
key
2023-11-23T15:14:38.577+0530 [INFO]  core: successfully setup
plugin catalog: plugin-directory=""
2023-11-23T15:14:38.577+0530 [INFO]  core: no mounts; adding
default mount table
2023-11-23T15:14:38.578+0530 [INFO]  core: successfully mounted:
type=cubbyhole version="v1.14.0+builtin.vault" path=cubbyhole/
namespace="ID: root. Path: "
2023-11-23T15:14:38.578+0530 [INFO]  core: successfully mounted:
type=system version="v1.14.0+builtin.vault" path=sys/
namespace="ID: root. Path: "
2023-11-23T15:14:38.578+0530 [INFO]  core: successfully mounted:
```

```
type=identity version="v1.14.0+builtin.vault" path=identity/
namespace="ID: root. Path: "
2023-11-23T15:14:38.581+0530 [INFO]  core: successfully mounted:
type=token version="v1.14.0+builtin.vault" path=token/
namespace="ID: root. Path: "
2023-11-23T15:14:38.581+0530 [INFO]  rollback: starting rollback
manager
2023-11-23T15:14:38.581+0530 [INFO]  core: restoring leases
2023-11-23T15:14:38.582+0530 [INFO]  expiration: lease restore
complete
2023-11-23T15:14:38.582+0530 [INFO]  identity: entities restored
2023-11-23T15:14:38.582+0530 [INFO]  identity: groups restored
2023-11-23T15:14:38.583+0530 [INFO]  core: Recorded vault
version: vault version=1.14.0 upgrade time="2023-11-23
09:44:38.582881162 +0000 UTC" build date=2023-06-19T11:40:23Z
2023-11-23T15:14:38.583+0530 [INFO]  core: usage gauge collection
is disabled
2023-11-23T15:14:38.998+0530 [INFO]  core: post-unseal setup
complete
2023-11-23T15:14:38.999+0530 [INFO]  core: root token generated
2023-11-23T15:14:38.999+0530 [INFO]  core: pre-seal teardown
starting
2023-11-23T15:14:38.999+0530 [INFO]  rollback: stopping rollback
manager
2023-11-23T15:14:38.999+0530 [INFO]  core: pre-seal teardown
complete
2023-11-23T15:14:39.311+0530 [INFO]  core.cluster-listener.tcp:
starting listener: listener_address=0.0.0.0:8201
2023-11-23T15:14:39.311+0530 [INFO]  core.cluster-listener:
serving cluster requests: cluster_listen_address=[::]:8201
2023-11-23T15:14:39.312+0530 [INFO]  core: post-unseal setup
starting
2023-11-23T15:14:39.312+0530 [INFO]  core: loaded wrapping token
key
2023-11-23T15:14:39.312+0530 [INFO]  core: successfully setup
plugin catalog: plugin-directory=""
2023-11-23T15:14:39.313+0530 [INFO]  core: successfully mounted:
type=system version="v1.14.0+builtin.vault" path=sys/
namespace="ID: root. Path: "
2023-11-23T15:14:39.313+0530 [INFO]  core: successfully mounted:
type=identity version="v1.14.0+builtin.vault" path=identity/
namespace="ID: root. Path: "
2023-11-23T15:14:39.313+0530 [INFO]  core: successfully mounted:
type=cubbyhole version="v1.14.0+builtin.vault" path=cubbyhole/
namespace="ID: root. Path: "
2023-11-23T15:14:39.314+0530 [INFO]  core: successfully mounted:
```

```
type=token version="v1.14.0+builtin.vault" path=token/
namespace="ID: root. Path: "
2023-11-23T15:14:39.314+0530 [INFO]  rollback: starting rollback
manager
2023-11-23T15:14:39.314+0530 [INFO]  core: restoring leases
2023-11-23T15:14:39.314+0530 [INFO]  identity: entities restored
2023-11-23T15:14:39.314+0530 [INFO]  expiration: lease restore
complete
2023-11-23T15:14:39.314+0530 [INFO]  identity: groups restored
2023-11-23T15:14:39.315+0530 [INFO]  core: usage gauge collection
is disabled
2023-11-23T15:14:39.316+0530 [INFO]  core: post-unseal setup
complete
2023-11-23T15:14:39.316+0530 [INFO]  core: vault is unsealed
Success! Uploaded policy: mcctb-policy
2023-11-23T15:14:39.795+0530 [INFO]  core: enabled credential
backend: path=approle/ type=approle version=""
Success! Enabled approle auth method at: approle/
2023-11-23T15:14:39.885+0530 [INFO]  core: successful mount:
namespace="" path=mcctb/ type=kv version=""
Success! Enabled the kv secrets engine at: mcctb/
Success! Data written to: auth/approle/role/mcctb-app
Installing the NetApp-MetroCluster-Tiebreaker-Software-1.6-
1.x86_64.rpm
Preparing...                          #
################################# # [100%]

Updating / installing...

1:NetApp-MetroCluster-Tiebreaker-So#
################################# # [100%]
Performing file integrity check
etc/cron.weekly/metrocluster-tiebreaker-support is Ok
etc/cron.weekly/metrocluster-tiebreaker-support-cov is Ok
etc/init.d/netapp-metrocluster-tiebreaker-software is Ok
etc/init.d/netapp-metrocluster-tiebreaker-software-cov is Ok
etc/logrotate.d/mcctb is Ok
opt/netapp/mcctb/lib/common/activation-1.1.1.jar is Ok
opt/netapp/mcctb/lib/common/aopalliance.jar is Ok
opt/netapp/mcctb/lib/common/args4j.jar is Ok
opt/netapp/mcctb/lib/common/aspectjrt.jar is Ok
opt/netapp/mcctb/lib/common/aspectjweaver.jar is Ok
opt/netapp/mcctb/lib/common/asup.jar is Ok
opt/netapp/mcctb/lib/common/bcpkix-jdk15on.jar is Ok
opt/netapp/mcctb/lib/common/bcprov-jdk15on.jar is Ok
opt/netapp/mcctb/lib/common/bcprov-jdk18on.jar is Ok
```

```
opt/netapp/mcctb/lib/common/bctls-fips-1.0.13.jar is Ok
opt/netapp/mcctb/lib/common/bctls-jdk18on.jar is Ok
opt/netapp/mcctb/lib/common/bcutil-jdk18on.jar is Ok
opt/netapp/mcctb/lib/common/cglib.jar is Ok
opt/netapp/mcctb/lib/common/commons-codec.jar is Ok
opt/netapp/mcctb/lib/common/commons-collections4.jar is Ok
opt/netapp/mcctb/lib/common/commons-compress.jar is Ok
opt/netapp/mcctb/lib/common/commons-daemon.jar is Ok
opt/netapp/mcctb/lib/common/commons-daemon.src.jar is Ok
opt/netapp/mcctb/lib/common/commons-dbcp2.jar is Ok
opt/netapp/mcctb/lib/common/commons-io.jar is Ok
opt/netapp/mcctb/lib/common/commons-lang3.jar is Ok
opt/netapp/mcctb/lib/common/commons-logging.jar is Ok
opt/netapp/mcctb/lib/common/commons-pool2.jar is Ok
opt/netapp/mcctb/lib/common/guava.jar is Ok
opt/netapp/mcctb/lib/common/httpclient.jar is Ok
opt/netapp/mcctb/lib/common/httpcore.jar is Ok
opt/netapp/mcctb/lib/common/jakarta.activation.jar is Ok
opt/netapp/mcctb/lib/common/jakarta.xml.bind-api.jar is Ok
opt/netapp/mcctb/lib/common/java-xmlbuilder.jar is Ok
opt/netapp/mcctb/lib/common/javax.inject.jar is Ok
opt/netapp/mcctb/lib/common/jaxb-api-2.3.1.jar is Ok
opt/netapp/mcctb/lib/common/jaxb-core.jar is Ok
opt/netapp/mcctb/lib/common/jaxb-impl.jar is Ok
opt/netapp/mcctb/lib/common/jline.jar is Ok
opt/netapp/mcctb/lib/common/jna.jar is Ok
opt/netapp/mcctb/lib/common/joda-time.jar is Ok
opt/netapp/mcctb/lib/common/jsch.jar is Ok
opt/netapp/mcctb/lib/common/json.jar is Ok
opt/netapp/mcctb/lib/common/jsvc.zip is Ok
opt/netapp/mcctb/lib/common/junixsocket-common.jar is Ok
opt/netapp/mcctb/lib/common/junixsocket-native-common.jar is Ok
opt/netapp/mcctb/lib/common/logback-classic.jar is Ok
opt/netapp/mcctb/lib/common/logback-core.jar is Ok
opt/netapp/mcctb/lib/common/mail-1.6.2.jar is Ok
opt/netapp/mcctb/lib/common/mariadb-java-client.jar is Ok
opt/netapp/mcctb/lib/common/mcctb-mib.jar is Ok
opt/netapp/mcctb/lib/common/mcctb.jar is Ok
opt/netapp/mcctb/lib/common/mockito-core.jar is Ok
opt/netapp/mcctb/lib/common/slf4j-api.jar is Ok
opt/netapp/mcctb/lib/common/snmp4j.jar is Ok
opt/netapp/mcctb/lib/common/spring-aop.jar is Ok
opt/netapp/mcctb/lib/common/spring-beans.jar is Ok
opt/netapp/mcctb/lib/common/spring-context-support.jar is Ok
opt/netapp/mcctb/lib/common/spring-context.jar is Ok
opt/netapp/mcctb/lib/common/spring-core.jar is Ok
```

```
opt/netapp/mcctb/lib/common/spring-expression.jar is Ok
opt/netapp/mcctb/lib/common/spring-web.jar is Ok
opt/netapp/mcctb/lib/common/vault-java-driver.jar is Ok
opt/netapp/mcctb/lib/common/xz.jar is Ok
opt/netapp/mcctb/lib/org.jacoco.agent-0.8.8-runtime.jar is Ok
opt/netapp/mcctb/bin/mcctb-asup-invoke is Ok
opt/netapp/mcctb/bin/mcctb_postrotate is Ok
opt/netapp/mcctb/bin/netapp-metrocluster-tiebreaker-software-cli
is Ok
/

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 1.6.
```

从Tieb破碎 机1.5升级到1.6

按照以下步骤将Tieber1.5软件版本升级到Tieber1.6。

步骤

a. 在运行以下命令 `[root@mcctb ~] #` 提示升级软件：

```
sh MetroClusterTiebreakerInstall-1.6
```

成功升级后，系统将显示以下输出：

```
Extracting the MetroCluster Tiebreaker installation/upgrade
archive
Install digest hash is Ok
Performing the MetroCluster Tiebreaker code signature check
Install code signature is Ok

Enter database user name : root
```

```
Please enter database password for root
Enter password:

Password updated successfully in the database.

Do you wish to generate your own public-private key pair for
encrypting audit log? [Y/N]: y
Generating public-private key pair...
Configuring Vault...
==> Vault shutdown triggered
2023-07-21T00:30:22.335+0530 [INFO]  core: marked as sealed
2023-07-21T00:30:22.335+0530 [INFO]  core: pre-seal teardown
starting
2023-07-21T00:30:22.335+0530 [INFO]  rollback: stopping rollback
manager
2023-07-21T00:30:22.335+0530 [INFO]  core: pre-seal teardown
complete
2023-07-21T00:30:22.335+0530 [INFO]  core: stopping cluster
listeners
2023-07-21T00:30:22.335+0530 [INFO]  core.cluster-listener:
forwarding rpc listeners stopped
2023-07-21T00:30:22.375+0530 [INFO]  core.cluster-listener: rpc
listeners successfully shut down
2023-07-21T00:30:22.375+0530 [INFO]  core: cluster listeners
successfully shut down
2023-07-21T00:30:22.376+0530 [INFO]  core: vault is sealed
Starting vault server...
==> Vault server configuration:

             Api Address: <api_address>
                     Cgo: disabled
         Cluster Address: <cluster_address>
   Environment Variables: BASH_FUNC_which%%,
DBUS_SESSION_BUS_ADDRESS, GODEBUG, HISTCONTROL, HISTSIZE, HOME,
HOSTNAME, HOST_ACCOUNT, LANG, LESSOPEN, LOGNAME, LS_COLORS, MAIL,
PATH, PWD, SHELL, SHLVL, SSH_CLIENT, SSH_CONNECTION, SSH_TTY,
STAF_TEMP_DIR, TERM, USER, VAULT_ADDR, VAULT_TOKEN,
XDG_RUNTIME_DIR, XDG_SESSION_ID, _, vault_Addr, which_declare
              Go Version: go1.20.5
              Listener 1: tcp (addr: "0.0.0.0:8200", cluster
address: "0.0.0.0:8201", max_request_duration: "1m30s",
max_request_size: "33554432", tls: "enabled")
               Log Level:
                   Mlock: supported: true, enabled: true
           Recovery Mode: false
```

```
                  Storage: file
                  Version: Vault v1.14.0, built 2023-06-
19T11:40:23Z
             Version Sha:
13a649f860186dffe3f3a4459814d87191efc321

==> Vault server started! Log data will stream in below:

2023-07-21T00:30:33.065+0530 [INFO]  proxy environment:
http_proxy="" https_proxy="" no_proxy=""
2023-07-21T00:30:33.098+0530 [INFO]  core: Initializing version
history cache for core
2023-07-21T00:30:43.092+0530 [INFO]  core: security barrier not
initialized
2023-07-21T00:30:43.092+0530 [INFO]  core: seal configuration
missing, not initialized
2023-07-21T00:30:43.094+0530 [INFO]  core: security barrier not
initialized
2023-07-21T00:30:43.096+0530 [INFO]  core: security barrier
initialized: stored=1 shares=5 threshold=3
2023-07-21T00:30:43.098+0530 [INFO]  core: post-unseal setup
starting
2023-07-21T00:30:43.124+0530 [INFO]  core: loaded wrapping token
key
2023-07-21T00:30:43.124+0530 [INFO]  core: successfully setup
plugin catalog: plugin-directory=""
2023-07-21T00:30:43.124+0530 [INFO]  core: no mounts; adding
default mount table
2023-07-21T00:30:43.125+0530 [INFO]  core: successfully mounted:
type=cubbyhole version="v1.14.0+builtin.vault" path=cubbyhole/
namespace="ID: root. Path: "
2023-07-21T00:30:43.126+0530 [INFO]  core: successfully mounted:
type=system version="v1.14.0+builtin.vault" path=sys/
namespace="ID: root. Path: "
2023-07-21T00:30:43.126+0530 [INFO]  core: successfully mounted:
type=identity version="v1.14.0+builtin.vault" path=identity/
namespace="ID: root. Path: "
2023-07-21T00:30:43.129+0530 [INFO]  core: successfully mounted:
type=token version="v1.14.0+builtin.vault" path=token/
namespace="ID: root. Path: "
2023-07-21T00:30:43.130+0530 [INFO]  rollback: starting rollback
manager
2023-07-21T00:30:43.130+0530 [INFO]  core: restoring leases
2023-07-21T00:30:43.130+0530 [INFO]  identity: entities restored
2023-07-21T00:30:43.130+0530 [INFO]  identity: groups restored
2023-07-21T00:30:43.131+0530 [INFO]  core: usage gauge collection
```

```
is disabled
2023-07-21T00:30:43.131+0530 [INFO]  expiration: lease restore
complete
2023-07-21T00:30:43.131+0530 [INFO]  core: Recorded vault
version: vault version=1.14.0 upgrade time="2023-07-20
19:00:43.131158543 +0000 UTC" build date=2023-06-19T11:40:23Z
2023-07-21T00:30:43.371+0530 [INFO]  core: post-unseal setup
complete
2023-07-21T00:30:43.371+0530 [INFO]  core: root token generated
2023-07-21T00:30:43.371+0530 [INFO]  core: pre-seal teardown
starting
2023-07-21T00:30:43.371+0530 [INFO]  rollback: stopping rollback
manager
2023-07-21T00:30:43.372+0530 [INFO]  core: pre-seal teardown
complete
2023-07-21T00:30:43.694+0530 [INFO]  core.cluster-listener.tcp:
starting listener: listener_address=0.0.0.0:8201
2023-07-21T00:30:43.695+0530 [INFO]  core.cluster-listener:
serving cluster requests: cluster_listen_address=[::]:8201
2023-07-21T00:30:43.695+0530 [INFO]  core: post-unseal setup
starting
2023-07-21T00:30:43.696+0530 [INFO]  core: loaded wrapping token
key
2023-07-21T00:30:43.696+0530 [INFO]  core: successfully setup
plugin catalog: plugin-directory=""
2023-07-21T00:30:43.697+0530 [INFO]  core: successfully mounted:
type=system version="v1.14.0+builtin.vault" path=sys/
namespace="ID: root. Path: "
2023-07-21T00:30:43.698+0530 [INFO]  core: successfully mounted:
type=identity version="v1.14.0+builtin.vault" path=identity/
namespace="ID: root. Path: "
2023-07-21T00:30:43.698+0530 [INFO]  core: successfully mounted:
type=cubbyhole version="v1.14.0+builtin.vault" path=cubbyhole/
namespace="ID: root. Path: "
2023-07-21T00:30:43.701+0530 [INFO]  core: successfully mounted:
type=token version="v1.14.0+builtin.vault" path=token/
namespace="ID: root. Path: "
2023-07-21T00:30:43.701+0530 [INFO]  rollback: starting rollback
manager
2023-07-21T00:30:43.702+0530 [INFO]  core: restoring leases
2023-07-21T00:30:43.702+0530 [INFO]  identity: entities restored
2023-07-21T00:30:43.702+0530 [INFO]  expiration: lease restore
complete
2023-07-21T00:30:43.702+0530 [INFO]  identity: groups restored
2023-07-21T00:30:43.702+0530 [INFO]  core: usage gauge collection
is disabled
```

```
2023-07-21T00:30:43.703+0530 [INFO]  core: post-unseal setup
complete
2023-07-21T00:30:43.703+0530 [INFO]  core: vault is unsealed
Success! Uploaded policy: mcctb-policy
2023-07-21T00:30:44.226+0530 [INFO]  core: enabled credential
backend: path=approle/ type=approle version=""
Success! Enabled approle auth method at: approle/
2023-07-21T00:30:44.315+0530 [INFO]  core: successful mount:
namespace="" path=mcctb/ type=kv version=""
Success! Enabled the kv secrets engine at: mcctb/
Success! Data written to: auth/approle/role/mcctb-app
Upgrading to NetApp-MetroCluster-Tiebreaker-Software-1.6-
1.x86_64.rpm
Preparing...
################################# [100%]
Updating / installing...
   1:NetApp-MetroCluster-Tiebreaker-
So################################# [ 50%]
Performing file integrity check
etc/cron.weekly/metrocluster-tiebreaker-support is Ok
etc/cron.weekly/metrocluster-tiebreaker-support-cov is Ok
etc/init.d/netapp-metrocluster-tiebreaker-software is Ok
etc/init.d/netapp-metrocluster-tiebreaker-software-cov is Ok
etc/logrotate.d/mcctb is Ok
opt/netapp/mcctb/lib/common/activation-1.1.1.jar is Ok
opt/netapp/mcctb/lib/common/aopalliance.jar is Ok
opt/netapp/mcctb/lib/common/args4j.jar is Ok
opt/netapp/mcctb/lib/common/aspectjrt.jar is Ok
opt/netapp/mcctb/lib/common/aspectjweaver.jar is Ok
opt/netapp/mcctb/lib/common/asup.jar is Ok
opt/netapp/mcctb/lib/common/bcpkix-jdk15on.jar is Ok
opt/netapp/mcctb/lib/common/bcprov-jdk15on.jar is Ok
opt/netapp/mcctb/lib/common/bcprov-jdk18on.jar is Ok
opt/netapp/mcctb/lib/common/bctls-fips-1.0.13.jar is Ok
opt/netapp/mcctb/lib/common/bctls-jdk18on.jar is Ok
opt/netapp/mcctb/lib/common/bcutil-jdk18on.jar is Ok
opt/netapp/mcctb/lib/common/cglib.jar is Ok
opt/netapp/mcctb/lib/common/commons-codec.jar is Ok
opt/netapp/mcctb/lib/common/commons-collections4.jar is Ok
opt/netapp/mcctb/lib/common/commons-compress.jar is Ok
opt/netapp/mcctb/lib/common/commons-daemon.jar is Ok
opt/netapp/mcctb/lib/common/commons-daemon.src.jar is Ok
opt/netapp/mcctb/lib/common/commons-dbcp2.jar is Ok
opt/netapp/mcctb/lib/common/commons-io.jar is Ok
opt/netapp/mcctb/lib/common/commons-lang3.jar is Ok
opt/netapp/mcctb/lib/common/commons-logging.jar is Ok
```

```
opt/netapp/mcctb/lib/common/commons-pool2.jar is Ok
opt/netapp/mcctb/lib/common/guava.jar is Ok
opt/netapp/mcctb/lib/common/httpclient.jar is Ok
opt/netapp/mcctb/lib/common/httpcore.jar is Ok
opt/netapp/mcctb/lib/common/jakarta.activation.jar is Ok
opt/netapp/mcctb/lib/common/jakarta.xml.bind-api.jar is Ok
opt/netapp/mcctb/lib/common/java-xmlbuilder.jar is Ok
opt/netapp/mcctb/lib/common/javax.inject.jar is Ok
opt/netapp/mcctb/lib/common/jaxb-api-2.3.1.jar is Ok
opt/netapp/mcctb/lib/common/jaxb-core.jar is Ok
opt/netapp/mcctb/lib/common/jaxb-impl.jar is Ok
opt/netapp/mcctb/lib/common/jline.jar is Ok
opt/netapp/mcctb/lib/common/jna.jar is Ok
opt/netapp/mcctb/lib/common/joda-time.jar is Ok
opt/netapp/mcctb/lib/common/jsch.jar is Ok
opt/netapp/mcctb/lib/common/json.jar is Ok
opt/netapp/mcctb/lib/common/jsvc.zip is Ok
opt/netapp/mcctb/lib/common/junixsocket-common.jar is Ok
opt/netapp/mcctb/lib/common/junixsocket-native-common.jar is Ok
opt/netapp/mcctb/lib/common/logback-classic.jar is Ok
opt/netapp/mcctb/lib/common/logback-core.jar is Ok
opt/netapp/mcctb/lib/common/mail-1.6.2.jar is Ok
opt/netapp/mcctb/lib/common/mariadb-java-client.jar is Ok
opt/netapp/mcctb/lib/common/mcctb-mib.jar is Ok
opt/netapp/mcctb/lib/common/mcctb.jar is Ok
opt/netapp/mcctb/lib/common/mockito-core.jar is Ok
opt/netapp/mcctb/lib/common/slf4j-api.jar is Ok
opt/netapp/mcctb/lib/common/snmp4j.jar is Ok
opt/netapp/mcctb/lib/common/spring-aop.jar is Ok
opt/netapp/mcctb/lib/common/spring-beans.jar is Ok
opt/netapp/mcctb/lib/common/spring-context-support.jar is Ok
opt/netapp/mcctb/lib/common/spring-context.jar is Ok
opt/netapp/mcctb/lib/common/spring-core.jar is Ok
opt/netapp/mcctb/lib/common/spring-expression.jar is Ok
opt/netapp/mcctb/lib/common/spring-web.jar is Ok
opt/netapp/mcctb/lib/common/vault-java-driver.jar is Ok
opt/netapp/mcctb/lib/common/xz.jar is Ok
opt/netapp/mcctb/bin/mcctb_postrotate is Ok
opt/netapp/mcctb/bin/netapp-metrocluster-tiebreaker-software-cli
is Ok
/

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 1.6.
Cleaning up / removing...
   2:NetApp-MetroCluster-Tiebreaker-
So################################# [100%]
```

从Tieb破碎 机1.4升级到1.6

按照以下步骤将Tieber1.4软件版本升级到Tieber1.6。

步骤

a. 在运行以下命令 `[root@mcctb ~] #` 提示升级软件：

```
sh MetroClusterTiebreakerInstall-1.6
```

成功升级后，系统将显示以下输出：

```
Extracting the MetroCluster Tiebreaker installation/upgrade
archive
Install digest hash is Ok
Performing the MetroCluster Tiebreaker code signature check
Install code signature is Ok
Enter unix user account to use for the installation:
mcctbuseradmin1
Unix user account "mcctbuseradmin1" doesn't exist. Do you wish to
create "mcctbuseradmin1" user account? [Y/N]: y
Unix account "mcctbuseradmin1" created.
Changing password for user mcctbuseradmin1.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.

Enter database user name : root

Please enter database password for root
Enter password:

Password updated successfully in the database.

MetroCluster Tiebreaker requires unix user account
```

```
"mcctbuseradmin1" to be added to the group "mcctbgrp" for admin
access.
Do you wish to add ? [Y/N]: y
Unix user account "mcctbuseradmin1" added to "mcctbgrp".
Do you wish to generate your own public-private key pair for
encrypting audit log? [Y/N]: y
Generating public-private key pair...
Configuring Vault...
Starting vault server...
==> Vault server configuration:

             Api Address: <api_addess>
                     Cgo: disabled
         Cluster Address: <cluster_address>
   Environment Variables: BASH_FUNC_which%%,
DBUS_SESSION_BUS_ADDRESS, GODEBUG, HISTCONTROL, HISTSIZE, HOME,
HOSTNAME, HOST_ACCOUNT, LANG, LESSOPEN, LOGNAME, LS_COLORS, MAIL,
PATH, PWD, SHELL, SHLVL, SSH_CLIENT, SSH_CONNECTION, SSH_TTY,
STAF_TEMP_DIR, TERM, USER, VAULT_ADDR, VAULT_TOKEN,
XDG_RUNTIME_DIR, XDG_SESSION_ID, _, vault_Addr, which_declare
              Go Version: go1.20.5
              Listener 1: tcp (addr: "0.0.0.0:8200", cluster
address: "0.0.0.0:8201", max_request_duration: "1m30s",
max_request_size: "33554432", tls: "enabled")
               Log Level:
                   Mlock: supported: true, enabled: true
           Recovery Mode: false
                 Storage: file
                 Version: Vault v1.14.0, built 2023-06-
19T11:40:23Z
             Version Sha:
13a649f860186dffe3f3a4459814d87191efc321

==> Vault server started! Log data will stream in below:

2023-11-23T15:58:10.400+0530 [INFO]  proxy environment:
http_proxy="" https_proxy="" no_proxy=""
2023-11-23T15:58:10.432+0530 [INFO]  core: Initializing version
history cache for core
2023-11-23T15:58:20.422+0530 [INFO]  core: security barrier not
initialized
2023-11-23T15:58:20.422+0530 [INFO]  core: seal configuration
missing, not initialized
2023-11-23T15:58:20.424+0530 [INFO]  core: security barrier not
initialized
2023-11-23T15:58:20.425+0530 [INFO]  core: security barrier
```

```
initialized: stored=1 shares=5 threshold=3
2023-11-23T15:58:20.427+0530 [INFO]  core: post-unseal setup
starting
2023-11-23T15:58:20.448+0530 [INFO]  core: loaded wrapping token
key
2023-11-23T15:58:20.448+0530 [INFO]  core: successfully setup
plugin catalog: plugin-directory=""
2023-11-23T15:58:20.448+0530 [INFO]  core: no mounts; adding
default mount table
2023-11-23T15:58:20.449+0530 [INFO]  core: successfully mounted:
type=cubbyhole version="v1.14.0+builtin.vault" path=cubbyhole/
namespace="ID: root. Path: "
2023-11-23T15:58:20.449+0530 [INFO]  core: successfully mounted:
type=system version="v1.14.0+builtin.vault" path=sys/
namespace="ID: root. Path: "
2023-11-23T15:58:20.449+0530 [INFO]  core: successfully mounted:
type=identity version="v1.14.0+builtin.vault" path=identity/
namespace="ID: root. Path: "
2023-11-23T15:58:20.451+0530 [INFO]  core: successfully mounted:
type=token version="v1.14.0+builtin.vault" path=token/
namespace="ID: root. Path: "
2023-11-23T15:58:20.452+0530 [INFO]  rollback: starting rollback
manager
2023-11-23T15:58:20.452+0530 [INFO]  core: restoring leases
2023-11-23T15:58:20.453+0530 [INFO]  identity: entities restored
2023-11-23T15:58:20.453+0530 [INFO]  identity: groups restored
2023-11-23T15:58:20.453+0530 [INFO]  expiration: lease restore
complete
2023-11-23T15:58:20.453+0530 [INFO]  core: usage gauge collection
is disabled
2023-11-23T15:58:20.453+0530 [INFO]  core: Recorded vault
version: vault version=1.14.0 upgrade time="2023-11-23
10:28:20.453481904 +0000 UTC" build date=2023-06-19T11:40:23Z
2023-11-23T15:58:20.818+0530 [INFO]  core: post-unseal setup
complete
2023-11-23T15:58:20.819+0530 [INFO]  core: root token generated
2023-11-23T15:58:20.819+0530 [INFO]  core: pre-seal teardown
starting
2023-11-23T15:58:20.819+0530 [INFO]  rollback: stopping rollback
manager
2023-11-23T15:58:20.819+0530 [INFO]  core: pre-seal teardown
complete
2023-11-23T15:58:21.116+0530 [INFO]  core.cluster-listener.tcp:
starting listener: listener_address=0.0.0.0:8201
2023-11-23T15:58:21.116+0530 [INFO]  core.cluster-listener:
serving cluster requests: cluster_listen_address=[::]:8201
```

```
2023-11-23T15:58:21.117+0530 [INFO]  core: post-unseal setup
starting
2023-11-23T15:58:21.117+0530 [INFO]  core: loaded wrapping token
key
2023-11-23T15:58:21.117+0530 [INFO]  core: successfully setup
plugin catalog: plugin-directory=""
2023-11-23T15:58:21.119+0530 [INFO]  core: successfully mounted:
type=system version="v1.14.0+builtin.vault" path=sys/
namespace="ID: root. Path: "
2023-11-23T15:58:21.120+0530 [INFO]  core: successfully mounted:
type=identity version="v1.14.0+builtin.vault" path=identity/
namespace="ID: root. Path: "
2023-11-23T15:58:21.120+0530 [INFO]  core: successfully mounted:
type=cubbyhole version="v1.14.0+builtin.vault" path=cubbyhole/
namespace="ID: root. Path: "
2023-11-23T15:58:21.123+0530 [INFO]  core: successfully mounted:
type=token version="v1.14.0+builtin.vault" path=token/
namespace="ID: root. Path: "
2023-11-23T15:58:21.123+0530 [INFO]  rollback: starting rollback
manager
2023-11-23T15:58:21.124+0530 [INFO]  core: restoring leases
2023-11-23T15:58:21.124+0530 [INFO]  identity: entities restored
2023-11-23T15:58:21.124+0530 [INFO]  identity: groups restored
2023-11-23T15:58:21.124+0530 [INFO]  expiration: lease restore
complete
2023-11-23T15:58:21.125+0530 [INFO]  core: usage gauge collection
is disabled
2023-11-23T15:58:21.125+0530 [INFO]  core: post-unseal setup
complete
2023-11-23T15:58:21.125+0530 [INFO]  core: vault is unsealed
Success! Uploaded policy: mcctb-policy
2023-11-23T15:58:21.600+0530 [INFO]  core: enabled credential
backend: path=approle/ type=approle version=""
Success! Enabled approle auth method at: approle/
2023-11-23T15:58:21.690+0530 [INFO]  core: successful mount:
namespace="" path=mcctb/ type=kv version=""
Success! Enabled the kv secrets engine at: mcctb/
Success! Data written to: auth/approle/role/mcctb-app
Upgrading to NetApp-MetroCluster-Tiebreaker-Software-1.6-
1.x86_64.rpm
Preparing...
################################# [100%]
Updating / installing...
   1:NetApp-MetroCluster-Tiebreaker-
So################################# [ 50%]
Performing file integrity check
```

```
etc/cron.weekly/metrocluster-tiebreaker-support is Ok
etc/cron.weekly/metrocluster-tiebreaker-support-cov is Ok
etc/init.d/netapp-metrocluster-tiebreaker-software is Ok
etc/init.d/netapp-metrocluster-tiebreaker-software-cov is Ok
etc/logrotate.d/mcctb is Ok
opt/netapp/mcctb/lib/common/activation-1.1.1.jar is Ok
opt/netapp/mcctb/lib/common/aopalliance.jar is Ok
opt/netapp/mcctb/lib/common/args4j.jar is Ok
opt/netapp/mcctb/lib/common/aspectjrt.jar is Ok
opt/netapp/mcctb/lib/common/aspectjweaver.jar is Ok
opt/netapp/mcctb/lib/common/asup.jar is Ok
opt/netapp/mcctb/lib/common/bcpkix-jdk15on.jar is Ok
opt/netapp/mcctb/lib/common/bcprov-jdk15on.jar is Ok
opt/netapp/mcctb/lib/common/bcprov-jdk18on.jar is Ok
opt/netapp/mcctb/lib/common/bctls-fips-1.0.13.jar is Ok
opt/netapp/mcctb/lib/common/bctls-jdk18on.jar is Ok
opt/netapp/mcctb/lib/common/bcutil-jdk18on.jar is Ok
opt/netapp/mcctb/lib/common/cglib.jar is Ok
opt/netapp/mcctb/lib/common/commons-codec.jar is Ok
opt/netapp/mcctb/lib/common/commons-collections4.jar is Ok
opt/netapp/mcctb/lib/common/commons-compress.jar is Ok
opt/netapp/mcctb/lib/common/commons-daemon.jar is Ok
opt/netapp/mcctb/lib/common/commons-daemon.src.jar is Ok
opt/netapp/mcctb/lib/common/commons-dbcp2.jar is Ok
opt/netapp/mcctb/lib/common/commons-io.jar is Ok
opt/netapp/mcctb/lib/common/commons-lang3.jar is Ok
opt/netapp/mcctb/lib/common/commons-logging.jar is Ok
opt/netapp/mcctb/lib/common/commons-pool2.jar is Ok
opt/netapp/mcctb/lib/common/guava.jar is Ok
opt/netapp/mcctb/lib/common/httpclient.jar is Ok
opt/netapp/mcctb/lib/common/httpcore.jar is Ok
opt/netapp/mcctb/lib/common/jakarta.activation.jar is Ok
opt/netapp/mcctb/lib/common/jakarta.xml.bind-api.jar is Ok
opt/netapp/mcctb/lib/common/java-xmlbuilder.jar is Ok
opt/netapp/mcctb/lib/common/javax.inject.jar is Ok
opt/netapp/mcctb/lib/common/jaxb-api-2.3.1.jar is Ok
opt/netapp/mcctb/lib/common/jaxb-core.jar is Ok
opt/netapp/mcctb/lib/common/jaxb-impl.jar is Ok
opt/netapp/mcctb/lib/common/jline.jar is Ok
opt/netapp/mcctb/lib/common/jna.jar is Ok
opt/netapp/mcctb/lib/common/joda-time.jar is Ok
opt/netapp/mcctb/lib/common/jsch.jar is Ok
opt/netapp/mcctb/lib/common/json.jar is Ok
opt/netapp/mcctb/lib/common/jsvc.zip is Ok
opt/netapp/mcctb/lib/common/junixsocket-common.jar is Ok
opt/netapp/mcctb/lib/common/junixsocket-native-common.jar is Ok
```

```
opt/netapp/mcctb/lib/common/logback-classic.jar is Ok
opt/netapp/mcctb/lib/common/logback-core.jar is Ok
opt/netapp/mcctb/lib/common/mail-1.6.2.jar is Ok
opt/netapp/mcctb/lib/common/mariadb-java-client.jar is Ok
opt/netapp/mcctb/lib/common/mcctb-mib.jar is Ok
opt/netapp/mcctb/lib/common/mcctb.jar is Ok
opt/netapp/mcctb/lib/common/mockito-core.jar is Ok
opt/netapp/mcctb/lib/common/slf4j-api.jar is Ok
opt/netapp/mcctb/lib/common/snmp4j.jar is Ok
opt/netapp/mcctb/lib/common/spring-aop.jar is Ok
opt/netapp/mcctb/lib/common/spring-beans.jar is Ok
opt/netapp/mcctb/lib/common/spring-context-support.jar is Ok
opt/netapp/mcctb/lib/common/spring-context.jar is Ok
opt/netapp/mcctb/lib/common/spring-core.jar is Ok
opt/netapp/mcctb/lib/common/spring-expression.jar is Ok
opt/netapp/mcctb/lib/common/spring-web.jar is Ok
opt/netapp/mcctb/lib/common/vault-java-driver.jar is Ok
opt/netapp/mcctb/lib/common/xz.jar is Ok
opt/netapp/mcctb/lib/org.jacoco.agent-0.8.8-runtime.jar is Ok
opt/netapp/mcctb/bin/mcctb-asup-invoke is Ok
opt/netapp/mcctb/bin/mcctb_postrotate is Ok
opt/netapp/mcctb/bin/netapp-metrocluster-tiebreaker-software-cli
is Ok
/

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 1.6.
Cleaning up / removing...
   2:NetApp-MetroCluster-Tiebreaker-
So################################# [100%]
```

安装Tieb破碎 锤1.5

配置对ONTAP API和SSH的管理员访问权限

您可以配置对ONTAP API和SSH的管理员访问权限。

步骤

1. 创建具有ONTAP API访问权限的管理员用户： `security login create -user-or-group-name mcctb -application ontapi -authentication-method password`
2. 创建具有SSH访问权限的管理员用户： `security login create -user-or-group-name mcctb -application ssh -authentication-method password`
3. 验证是否已创建新的管理员用户： `security login show`
4. 在配对集群上重复上述步骤。

> "管理员身份验证和 RBAC" 已实施。

安装MetroCluster Tieber1.5依赖关系

根据您的主机Linux操作系统、您必须先安装MySQL或MariaDB服务器、然后再安装或升级Tieb破碎 机软件。

步骤

1. 安装JDK
2. 安装和配置存储
3. 安装 MySQL 或 MariaDB 服务器：

如果 Linux 主机为	那么 …
Red Hat Enterprise Linux 7/CentOS 7.	在Red Hat Enterprise Linux 7或CentOS 7上安装MySQL Server 5.5.30或更高版本以及5.6.x版本
Red Hat Enterprise Linux 8	在Red Hat Enterprise Linux 8上安装MariaDB服务器

安装JDK

在安装或升级Tieb破碎 机软件之前、您必须在主机系统上安装JDK。Tieber1.5及更高版本支持OpenJDK 17、18或19。

步骤

1. 以"root"用户或可更改为高级权限模式的sudo用户身份登录。

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

2. 检查可用的JDK版本：

   ```
   yum search openjdk
   ```

3. 安装JDK 17、18或19。

   以下命令将安装JDK 17：

   ```
   yum install java-17-openjdk
   ```

4. 验证安装。

   ```
   java -version
   ```

   成功安装将显示以下输出：

   ```
   openjdk version "17.0.2" 2022-01-18 LTS
   OpenJDK Runtime Environment 21.9 (build 17.0.2+8-LTS)
   OpenJDK 64-Bit Server VM 21.9 (build 17.0.2+8-LTS, mixed mode, sharing)
   ```

安装和配置存储

如果您没有或不想使用本地存储服务器、则必须安装存储。您可以参考此标准操作步骤 来安装存储、也可以参考Hashicorp安装说明来了解其他准则。

> 如果网络中有存储服务器、则可以将MetroCluster Tiebreaker主机配置为使用该存储安装。如果执行此操作、则不需要在主机上安装存储。

步骤

1. 导航到 `/bin` 目录：

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

2. 下载存储zip文件。

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

3. 解压缩存储文件。

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

4. 验证安装。

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

5. 导航到 `/root` 目录：

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

6. 在下创建存储配置文件 `/root` 目录。

   在 `[root@mcctb ~]` 提示符下、复制并运行以下命令以创建 `config.hcl` 文件：

```
# 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. 启动存储服务器：

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

8. 导出存储地址。

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

9. 初始化存储。

```
[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.
```

您必须将密钥ID和初始根令牌记录并存储在一个安全位置、以供日后在操作步骤中使用。

10. 导出存储根令牌。

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

11. 使用创建的五个密钥中的任意三个来打开存储。

    您必须运行 `vault operator unseal` 命令：

    a. 使用第一个密钥打开存储：

    ```
    [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. 使用第二个密钥打开存储：

```
[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. 使用第三个密钥打开存储：

```
[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. 验证存储密封状态是否为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. 将存储服务配置为在引导时启动。
    a. 运行以下命令： `cd /etc/systemd/system`

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

    b. 在 `[root@mcctb system]` 提示符下、复制并运行以下命令以创建存储服务文件。

    ```
    # 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. 运行以下命令： `systemctl daemon-reload`

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

    d. 运行以下命令： `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.
```

在安装MetroCluster Tiebreaker期间、系统会提示您使用此功能。如果要更改此方法以取消存储密封、则需要卸载并重新安装MetroCluster Tiebreaker软件。

在Red Hat Enterprise Linux 7或CentOS 7上安装MySQL Server 5.5.30或更高版本以及5.6.x版本

在安装或升级 Tiebreaker 软件之前，必须在主机系统上安装 MySQL Server 5.5.30 或更高版本以及 5.6.x 版本。对于Red Hat Enterprise Linux 8、 安装MariaDB服务器。

步骤

1. 以root用户或可更改为高级权限模式的sudo用户身份登录。

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

2. 将 MySQL 存储库添加到主机系统：

` 根@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. 禁用MySQL 57存储库：

` 根@mcctb ~ ]# yam-config-manager -disable mysql57-community`

4. 启用MySQL 56存储库：

` 根@mcctb ~ ]# yam-config-manager -enable mysql56-community`

5. 启用存储库：

` 根@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. 安装 MySQL 社区服务器：

` 根@mcctb ~ ]# yum install mysql-commune-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. 启动 MySQL 服务器：

` 根@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. 确认 MySQL 服务器正在运行：

` 根@mcctb ~ ]# service mysqld status`

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

9. 配置安全性和密码设置：

` 根@mcctb ~ ]# mysql_secure 安装`

```
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. 验证 MySQL 登录是否正常工作：

` 根@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>
```

如果 MySQL 登录正常，输出将在 `mysql>` 提示符处结束。

启用**MySQL**自动启动设置

您应验证是否已为MySQL守护进程启用自动启动功能。如果 MetroCluster Tiebreaker 软件所在的系统重新启动，则打开 MySQL 守护进程会自动重新启动 MySQL 。如果 MySQL 守护进程未运行， Tiebreaker 软件将继续运行，但无法重新启动，并且无法更改配置。

步骤

1. 验证是否已启用 MySQL 在启动时自动启动：

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

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

   如果在启动时未启用 MySQL 自动启动，请参见 MySQL 文档为您的安装启用自动启动功能。

在Red Hat Enterprise Linux 8上安装MariaDB服务器

在安装或升级 Tiebreaker 软件之前，必须在主机系统上安装 MariaDB 服务器。对于Red Hat Enterprise Linux 7或CentOS 7、 安装MySQL Server。

开始之前

主机系统必须运行在 Red Hat Enterprise Linux （ RHEL ） 8 上。

步骤

1. 以登录身份 `root` 可通过sudo进入高级权限模式的用户。

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

2. 安装MariaDB服务器：

   ` 根@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!
```

3. 启动 MariaDB 服务器：

   ` 根@mcctb ~ ]# systemctl start MariaDB`

4. 验证MariaDB服务器是否已启动：

   ` 根@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.
```

5. 配置安全性和密码设置：

> 当系统提示您输入root密码时、请将其留空、然后按Enter继续配置安全和密码设置。

` 根@mcctb ~ ]# mysql_secure 安装`

```
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!
```

为**MariaDB**服务器启用自动启动设置

您应验证是否已为MariaDB服务器启用自动启动功能。如果不启用自动启动功能，并且 MetroCluster Tiebreaker 软件所在的系统必须重新启动，则 Tiebreaker 软件将继续运行，但无法重新启动 MariaDB 服务，也无法更改配置。

步骤

1. 启用自动启动服务：

   ` 根@mcctb ~ ]# systemctl enable mariadb.service`

2. 验证启动时 MariaDB 是否已启用自动启动：

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

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

安装或升级到**Tieb**破碎 机**1.5**

在主机Linux操作系统上全新安装或升级到Tieb破碎 机1.5、以监控MetroCluster配置。

关于此任务

- 存储系统必须运行受支持的ONTAP版本。请参见 "软件要求" 表以了解更多详细信息。
- 您必须已使用安装OpenJDK `yum install java-x.x.x-openjdk` 命令：Tieber1.5及更高版本支持OpenJDK 17、18或19。
- 您可以使用具有足够管理权限的非root用户身份安装MetroCluster Tieb破碎 镐、以便执行Tieb破碎 镐安装、创建表和用户以及设置用户密码。

步骤

1. 下载MetroCluster Tieb破碎 机软件和MetroCluster_Tieb破碎 机_RPM_GPG密钥。

   可以从NetApp 支持站点 上下载Tieb破碎 机1.5软件包的同一页面下载MetroCluster_Tieb破碎 机_RPM_GPG密钥。

   "MetroCluster Tieb破碎 机(下载)—NetApp 支持站点"

2. 以 root 用户身份登录到主机。
3. 创建非root用户和 `mcctbgrp` 组。

   a. 创建非root用户并设置密码。

      以下示例命令将创建一个名为的非root用户 `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. 创建一个名为的组 `mcctbgrp`：

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

   c. 将您创建的非root用户添加到 `mcctbgrp` 组。

      以下命令将添加 `mcctbuser1` 到 `mcctbgrp` 组：

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

4. 验证RPM文件。

   从包含RPM密钥的目录运行以下子步骤。

   a. 下载并导入RPM密钥文件：

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

   b. 通过检查指纹来验证是否导入了正确的密钥。

   以下示例显示了正确的密钥指纹：

```
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>
```

   a. 验证签名： `rpm --checksig NetApp-MetroCluster-Tiebreaker-Software-1.5-1.x86_64.rpm`

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

   > 只有在成功验证签名后、才能继续安装。

5. 【安装- Tiebreaker】安装或升级Tiebreaker软件：

   > 只有在从Tiebreaker 1.4版升级时、才能升级到Tiebreaker 1.5版。不支持从早期版本升级到Tiebreaker 1.5。

   根据您是执行新安装还是升级现有安装、选择正确的操作步骤。

执行新安装

a. 检索并记录Java的绝对路径：

```
[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. 运行以下命令： `rpm -ivh NetApp-MetroCluster-Tiebreaker-Software-1.5-1.x86_64.rpm`

成功安装时，系统将显示以下输出：

> 在安装期间出现提示时、请提供您先前创建并分配给的非root用户 `mcctbgrp` 组。

```
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.
```

升级现有安装

a. 验证是否已安装受支持的OpenJDK版本、以及是否为主机上的当前Java版本。

> 要升级到Tiebreaker 1.5、您必须安装OpenJDK 17、18或19版。

```
[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. 验证存储服务是否已取消密封并正在运行： `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. 升级Tiebreaker软件。

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

成功升级后，系统将显示以下输出：

```
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%]
```

> 如果输入的 MySQL root 密码不正确， Tiebreaker 软件会指示已成功安装该密码，但会显示 Access Denied 消息。要解决问题描述问题，您必须使用 `rpm -e` 命令卸载 Tiebreaker 软件，然后使用正确的 MySQL root 密码重新安装该软件。

6. 通过打开从Tiebreaker主机到每个节点管理LIF和集群管理LIF的SSH连接、检查Tiebreaker与MetroCluster软件的连接。

相关信息

"NetApp 支持"

安装Tieb破碎 锤1.4

安装MetroCluster Tieb破碎 机1.4依赖关系

根据您的主机Linux操作系统、在安装或升级Tieb破碎 机软件之前安装MySQL或MariaDB服务器。

步骤

1. 安装JDK。
2. 安装 MySQL 或 MariaDB 服务器：

如果 Linux 主机为	那么 …
Red Hat Enterprise Linux 7/CentOS 7.	在Red Hat Enterprise Linux 7或CentOS 7上安装MySQL Server 5.5.30或更高版本以及5.6.x版本
Red Hat Enterprise Linux 8	在Red Hat Enterprise Linux 8上安装MariaDB服务器

安装JDK

在安装或升级Tieb破碎 机软件之前、您必须在主机系统上安装JDK。Tieb破碎 机1.4及更早版本支持JDK 1.0.0。(JRE 8)。

步骤

1. 以"root"用户身份登录。

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

2. 安装JDK 1.0.0：

```
yum install java-1.8.0-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!
```

在Red Hat Enterprise Linux 7或CentOS 7上安装MySQL Server 5.5.30或更高版本以及5.6.x版本

在安装或升级 Tiebreaker 软件之前，必须在主机系统上安装 MySQL Server 5.5.30 或更高版本以及 5.6.x 版本。 对于Red Hat Enterprise Linux 8、 安装MariaDB服务器。

步骤

1. 以 root 用户身份登录。

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

2. 将 MySQL 存储库添加到主机系统：

` 根@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. 禁用MySQL 57存储库：

` 根@mcctb ~ ]# yam-config-manager -disable mysql57-community`

4. 启用MySQL 56存储库：

` 根@mcctb ~ ]# yam-config-manager -enable mysql56-community`

5. 启用存储库：

` 根@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. 安装 MySQL 社区服务器：

` 根@mcctb ~ ]# yum install mysql-commune-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. 启动 MySQL 服务器：

` 根@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. 确认 MySQL 服务器正在运行：

` 根@mcctb ~ ]# service mysqld status`

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

9. 配置安全性和密码设置：

` 根@mcctb ~ ]# mysql_secure 安装`

```
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. 验证 MySQL 登录是否正常工作：

` 根@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>
```

当MySQL登录按预期运行时、输出将在结束 `mysql>` 提示符。

启用**MySQL**自动启动设置

您应验证是否已为MySQL守护进程启用自动启动功能。如果 MetroCluster Tiebreaker 软件所在的系统重新启动，则打开 MySQL 守护进程会自动重新启动 MySQL 。如果 MySQL 守护进程未运行， Tiebreaker 软件将继续运行，但无法重新启动，并且无法更改配置。

步骤

1. 验证是否已启用 MySQL 在启动时自动启动：

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

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

   如果在启动时未启用 MySQL 自动启动，请参见 MySQL 文档为您的安装启用自动启动功能。

在Red Hat Enterprise Linux 8上安装MariaDB服务器

在安装或升级 Tiebreaker 软件之前，必须在主机系统上安装 MariaDB 服务器。 对于Red Hat Enterprise Linux 7或CentOS 7、 安装MySQL Server。

开始之前

主机系统必须运行在 Red Hat Enterprise Linux （ RHEL ） 8 上。

步骤

1. 以登录身份 `root` 用户。

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

2. 安装MariaDB服务器：

   ` 根@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!
```

3. 启动 MariaDB 服务器：

   ` 根@mcctb ~ ]# systemctl start MariaDB`

4. 验证MariaDB服务器是否已启动：

   ` 根@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.
```

5. 配置安全性和密码设置：

> 当系统提示您输入root密码时、请将其留空、然后按Enter继续配置安全和密码设置。

` 根@mcctb ~ ]# mysql_secure 安装`

```
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!
```

为**MariaDB**服务器启用自动启动设置

您应验证是否已为MariaDB服务器启用自动启动功能。如果不启用自动启动功能，并且 MetroCluster Tiebreaker 软件所在的系统必须重新启动，则 Tiebreaker 软件将继续运行，但无法重新启动 MariaDB 服务，也无法更改配置。

步骤

1. 启用自动启动服务：

   ` 根@mcctb ~ ]# systemctl enable mariadb.service`

2. 验证启动时 MariaDB 是否已启用自动启动：

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

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

安装或升级到**Tieb**破碎 机**1.4**

在主机Linux操作系统上全新安装或升级到Tieb破碎 机1.4、以监控MetroCluster配置。

关于此任务

- 存储系统必须运行受支持的ONTAP版本。请参见 "软件要求" 表以了解更多详细信息。
- 您必须已使用安装OpenJDK `yum install java-x.x.x-openjdk` 命令：TiebreAKER 1.4及更早版本支持JDK 1.8.0 (JRE 8)。

步骤

1. 下载MetroCluster Tieb破碎 机软件。

   "MetroCluster Tieb破碎 机(下载)—NetApp 支持站点"

2. 以 root 用户身份登录到主机。
3. 【安装- Tiebreaker】安装或升级Tiebreaker软件：

   根据您是执行新安装还是升级现有安装、选择正确的操作步骤。

执行新安装

a. 通过运行来安装Tieb破碎 机软件：

```
rpm -ivh NetApp-MetroCluster-Tiebreaker-Software-1.4-1.x86_64.rpm
```

成功安装时，系统将显示以下输出：

```
Verifying...
################################# [100%]
Preparing...
################################# [100%]
Updating / installing...
   1:NetApp-MetroCluster-Tiebreaker-
So################################# [100%]
Post installation start Fri Apr  5 02:28:09 EDT 2024
Enter MetroCluster Tiebreaker user password:

Please enter mysql root password when prompted
Enter password:
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
Enabled autostart of NetApp MetroCluster Tiebreaker software
daemon during boot
Created symbolic link for NetApp MetroCluster Tiebreaker software
CLI
Post installation end Fri Apr  5 02:28:22 EDT 2024
Successfully installed NetApp MetroCluster Tiebreaker software
version 1.4.
```

升级现有安装

a. 升级Tiebreaker软件。

```
[root@mcctb ~]# rpm -Uvh NetApp-MetroCluster-Tiebreaker-Software-
1.4-1.x86_64.rpm
```

成功升级后，系统将显示以下输出：

```
Verifying...
################################# [100%]
Preparing...
################################# [100%]
Upgrading NetApp MetroCluster Tiebreaker software....
Stopping NetApp MetroCluster Tiebreaker software services before
upgrade.
Updating / installing...
   1:NetApp-MetroCluster-Tiebreaker-
So################################## [ 50%]
Post installation start Mon Apr  8 06:29:51 EDT 2024
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
Enabled autostart of NetApp MetroCluster Tiebreaker software
daemon during boot
Created symbolic link for NetApp MetroCluster Tiebreaker software
CLI
Post upgrade end Mon Apr  8 06:29:51 EDT 2024
Successfully upgraded NetApp MetroCluster Tiebreaker software to
version 1.4.
Cleaning up / removing...
   2:NetApp-MetroCluster-Tiebreaker-
So################################## [100%]
```

> 如果输入的 MySQL root 密码不正确， Tiebreaker 软件会指示已成功安装该密码，但会显示 Access Denied 消息。要解决问题描述问题，您必须使用 `rpm -e` 命令卸载 Tiebreaker 软件，然后使用正确的 MySQL root 密码重新安装该软件。

4. 通过打开从Tiebreaker主机到每个节点管理LIF和集群管理LIF的SSH连接、检查Tiebreaker与MetroCluster软件的连接。

相关信息

"NetApp 支持"