

[向量数据库安装文档]

版本号[1.00]

文本

描述已自动生成

**2024-03-08**

广州云趣信息科技有限公司

广东省广州市天河区建工路4号

**版本修订批准记录**

【历次版本修订记录，版本修订描述重要说明】

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **提交日期** | **版本号** | **版本修订描述** | **作者** | **批准人** |
| 2024-2-20 | 1.00 | 初始版本 | 邓文静 |  |
| 2025-2-21 | 1.10 | 修改安装步骤，增加常见问题 | 邓文静 |  |
| 2025-3-3 | 1.20 | 1. 增加ARM版安装步骤 2. 删除非docker版安装步骤 | 邓文静 |  |
| 2025-3-21 | 1.30 | 修改安装步骤 | 邓文静 |  |
| 2025-3-31 | 1.40 | 增加集群版底层引擎安装步骤 | 邓文静 |  |
|  |  |  |  |  |

**注：版本修订批准记录，如不需要可删除此页**

**目录**

[1. 声明 4](#_Toc4)

[1.1 关于版权 4](#_Toc31581)

[1.2 阅读约定 5](#_Toc10452)

[1.3 编写目的 6](#_Toc29598)

[2. 部署前准备 6](#_Toc17573)

[2.1 安装包简介 6](#_Toc31884)

[2.2 硬件环境 6](#_Toc22093)

[2.3 软件环境 7](#_Toc32088)

[2.4 开放的端口 7](#_Toc323)

[3. 安装步骤 7](#_Toc2196)

[3.1 安装单机版底层引擎 7](#_Toc32472)

[3.2 安装集群版底层引擎 8](#_Toc26388)

[3.3 安装向量数据库 11](#_Toc15929)

[4. 常见问题 17](#_Toc30317)

[4.1 停止单机版底层引擎容器 17](#_Toc3414)

[4.2 删除数据库数据 17](#_Toc24762)

[4.3 升级openssl后yum无法使用 17](#_Toc23074)

[4.4 修改向量数据库端口 18](#_Toc8027)

[4.5 使用docker-compose创建容器失败 19](#_Toc22226)

# 声明

## 关于版权

广州云趣信息科技有限公司版权所有，受中国及其他国家商业法律及版权法保护，保留一切权利。未经本公司书面许可，任何单位和个人不得擅自摘抄、复制本书内容的部分或全部，并不得以任何形式传播。

我们尽力保证本文档的完整性和准确性，但不排除文档中有错误存在。广州云趣信息科技有限公司有权在不通知您的情况下更改此文档。

文本

描述已自动生成、“YQ”、“yunqu”均为广州云趣信息科技有限公司的商标。对于本手册中出现的其它公司的商标、产品标识及商品名称，由各自权利人拥有。

除非另有约定，本手册仅作为使用指导，本手册中的所有陈述、信息和建议不构成任何明示或暗示的担保。如需要获取最新手册，请登录http://www.yunqu-info.com。

## 阅读约定

|  |  |
| --- | --- |
| **格式** | **意义** |
| **加粗字体** | 手册中，所有加粗的正文字体均表示关键字段。 |
| **红色字体** | 手册中，所有红色显示字体，均表示需要重点、需要特别关注的字段。 |
| **< >** | 手册中，所有带“< >”均表示按钮名，如“单击<确定> 按钮”。 |
| **[ ]** | 带方括号“[ ]”表示窗口名、菜单名和数据表，如“弹出[新建用户]窗口”。 |
| **/** | 多级菜单及并行选项用“/”隔开。如[文件/新建/文件夹]多级菜单表示[文件]菜单下的[新建]子菜单下的[文件夹]菜单项。 |

## 编写目的

用于指导工程、测试人员安装部署向量数据库、进行简单的自检，以及常见问题的定位及解决。

# 部署前准备



## 安装包简介

安装包列表如下：

|  |  |  |
| --- | --- | --- |
| 参数包 | 安装服务器 | 安装包描述 |
| milvus\_docker.tar | 引擎服务器 | 单机版底层引擎镜像包 |
| run\_engine.sh | 引擎服务器 | 单机版底层引擎服务启动脚本 |
| milvus\_cluster\_images.tar.gz | 引擎服务器 | 集群版底层引擎镜像安装包 |
| milvus\_cluster.tar.gz | 引擎服务器 | 集群版底层引擎安装包 |
| vectordb.tar.gz | 引擎服务器 | 向量数据库安装包 |
| vectordb\_docker.tar | 引擎服务器 | 向量数据库镜像包 |

## 硬件环境

|  |  |
| --- | --- |
| 操作系统 | 操作系统为centos7.6 |
| 服务器配置 | CPU 8核32G以上；  CPU支持SSE4.2、AVX、AVX2、AVX-512其中一个指令集；  磁盘空间500GB及以上； |

## 软件环境

|  |  |
| --- | --- |
| 环境 | 状态 |
| Docker | docker已安装调试就绪（建议使用Docker版本为19.03及以上；公司版本服务器：[http://172.16.70.246/version/5#--配套第三方软件/Docker/Docker Engine/](http://172.16.70.246/version/5" \l "--配套第三方软件/Docker/Docker Engine/) 路径下） |
| docker-compose | docker-compose已经准备就绪（使用docker-compose.1.29或Docker Compose V2（建议）），公司版本服务器[http://172.16.70.246/version/5#--配套第三方软件/Docker/Docker Compose](http://172.16.70.246/version/5" \l "--配套第三方软件/Docker/Docker Compose)路径下 |
| Python | Python已经准备就绪，Python版本为3.9.12 |

## 开放的端口

|  |  |
| --- | --- |
| 服务 | 端口 |
| 底层引擎 | 2379、9091、19530 |
| 向量数据库 | 默认为8092，可按需修改 |

# 安装步骤



## 安装单机版底层引擎

**第一步：新建/home/yunqu/images并上传底层引擎镜像包到该目录**

mkdir -p /home/yunqu/images && cd /home/yunqu/images

**第二步：加载底层引擎镜像**

docker load -i milvus\_docker\_v2.5.6.tar

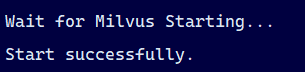
**第三步：新建底层引擎目录并上传run\_engine.sh到该目录**

mkdir -p /home/yunqu/data/milvus && cd /home/yunqu/data/milvus

**第四步：启动底层引擎服务**

. run\_engine.sh start

如果显示以下信息，则表示底层引擎启动成功：



## 安装集群版底层引擎

**第一步：上传集群版底层引擎镜像安装包到/home/yunqu/images并解压**

cd /home/yunqu/images

tar -zxvf milvus\_cluster\_images.tar.gz

**第二步：加载相关镜像**

for file in /home/yunqu/images/cluster\_images/\*.tar; do docker load -i "$file"; done

**第三步：上传集群版底层引擎安装包到/home/yunqu/data并解压**

cd /home/yunqu/data

tar -zxvf milvus\_cluster.tar.gz

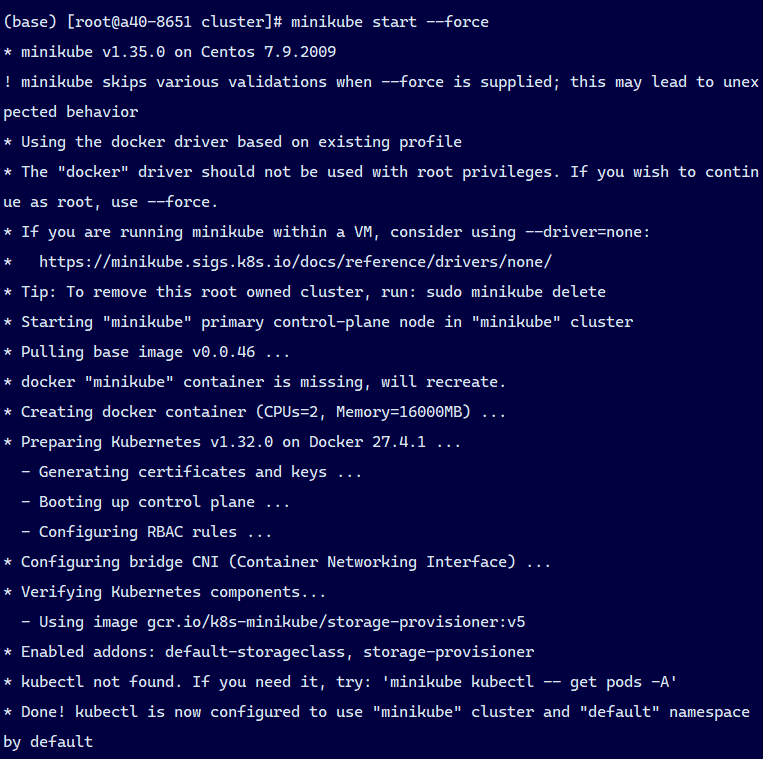
**第四步：安装并启动minikube**

cd /home/yunqu/data/milvus\_cluster

sudo install minikube-linux-amd64 /usr/local/bin/minikube

tar xf minikube\_cache.tar.gz -C /root/.minikube/

minikube start --force

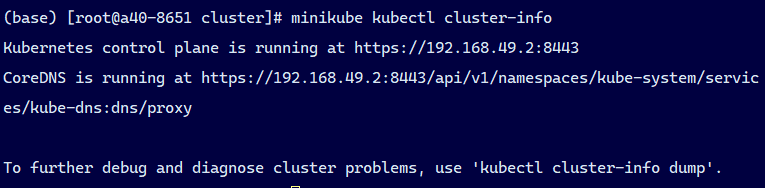


**第五步：检查k8s状态**

查询k8s集群的核心组件状态和访问地址

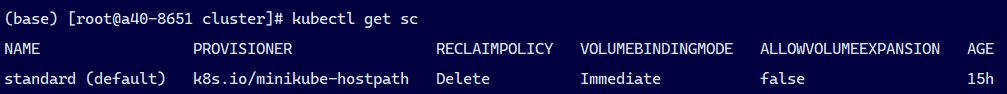
ln -s $(which minikube) /usr/local/bin/kubectl

kubectl cluster-info



检查k8s集群中可用的StorageClass

kubectl get sc

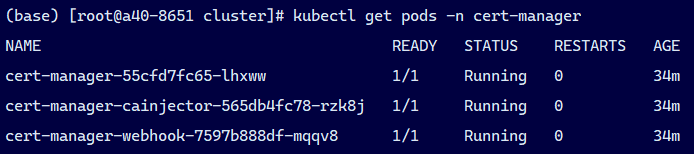


**第六步：安装cert-manager**

kubectl apply -f cert-manager.yaml

kubectl get pods -n cert-manager

如果显示以下信息，则表示cert-manager安装成功：

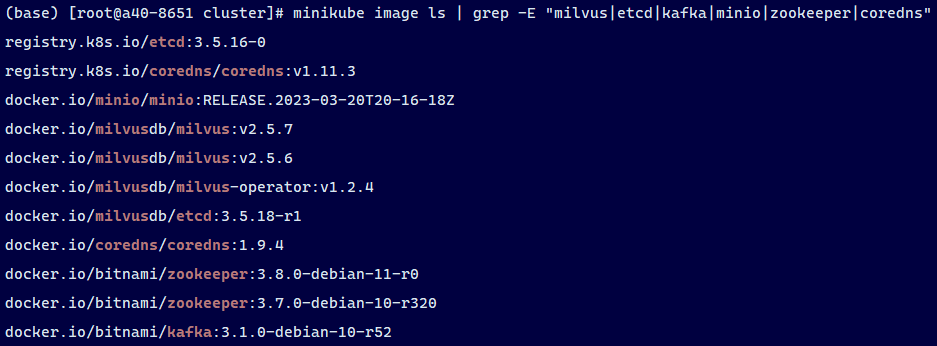


**第七步：将相关镜像导入minikube**

for img in images/\*.tar; do minikube image load "$img"; done

minikube image ls | grep -E "milvus|etcd|kafka|minio|zookeeper|coredns"

如果显示以下信息，则表示镜像加载成功：

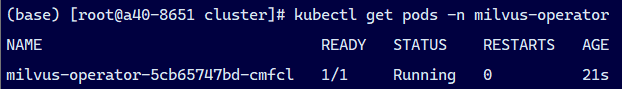


**第八步：安装milvus operator**

kubectl apply -f milvus-operator-deployment.yaml

kubectl get pods -n milvus-operator

如果显示以下信息，则表示milvus operator安装成功：



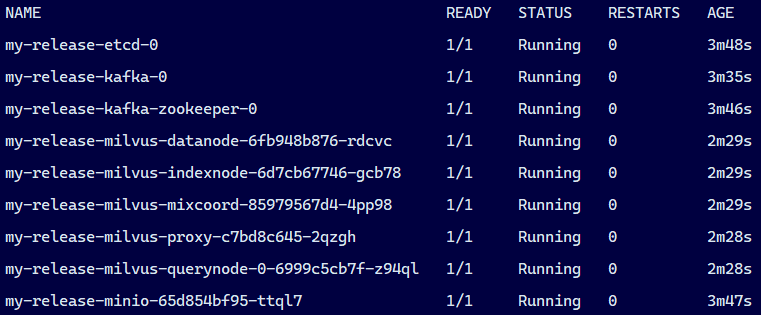
**第九步：部署milvus集群组件和依赖组件**

kubectl apply -f operatorConfigYml.yml

检查milvus集群中所有pod的状态

kubectl get pods

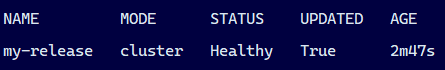
如果**READY**均显示**1/1**，且**STATUS**均显示**Running**，则表示milvus集群中的所有pod都处于运行状态：



检查milvus集群状态

kubectl get milvus my-release

如果显示以下信息，则表示milvus集群启动成功：



**第十步：将本地端口转发到服务端口**

kubectl port-forward --address 0.0.0.0 service/my-release-milvus 19530:19530 > /dev/null 2>&1 &

## 安装向量数据库

**第一步：新建/home/yunqu/data并上传向量数据库镜像包与安装包到该目录**

mkdir -p /home/yunqu/data && cd /home/yunqu/data

**第二步：加载向量数据库镜像**

docker load -i vectordb\_docker.tar

**第三步：解压向量数据库安装包**

tar -zxvf vectordb.tar.gz

**第四步：修改文件权限**

chmod 777 /home/yunqu/data/vectordb/start.sh

**第五步：修改配置文件**

修改milvus\_config.py文件中的**MILVUS\_HOST**为实际IP地址

vi /home/yunqu/data/vectordb/vectordb/app/backend/milvus\_config.py



（可选）修改configs.py文件中的端口号，默认端口号为8092，进程数为1，可按需修改

vi /home/yunqu/data/vectordb/app/plugins/configs.py



**第六步：修改服务启动脚本**

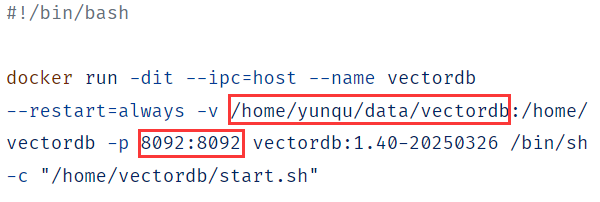
如果安装目录与端口号有变动，需同步修改脚本中的路径与端口号

# CPU版

vi run\_docker.sh

# GPU版

vi run\_docker\_gpu.sh



**第七步：启动向量数据库服务**

# CPU版

. run\_docker.sh

# GPU版

. run\_docker\_gpu.sh

启动服务需等待1~3分钟

**第八步：验证底层引擎服务**

修改check\_engine.py中的**"localhost"**为实际IP

vi check\_engine.py



运行底层引擎服务验证脚本

docker exec vectordb sh /home/vectordb/check\_engine.sh

如果显示如下信息，则表示服务正常运行：

IMG_256

**第九步：验证向量数据库服务**

如果端口号有改动，需要修改check\_api.py

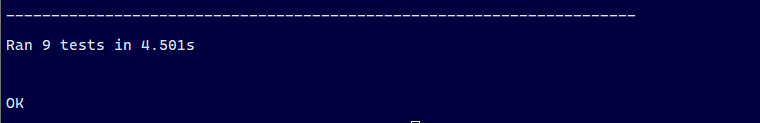
vi check\_api.py



运行向量数据库服务验证脚本

docker exec vectordb sh /home/vectordb/check\_api.sh

如果出现以下信息，则表示服务正常运行



# 常见问题



## 停止单机版底层引擎容器

docker stop milvus-standalone

## 删除数据库数据

如果需要在停止底层引擎后删除全部数据，需要删除volumes文件夹

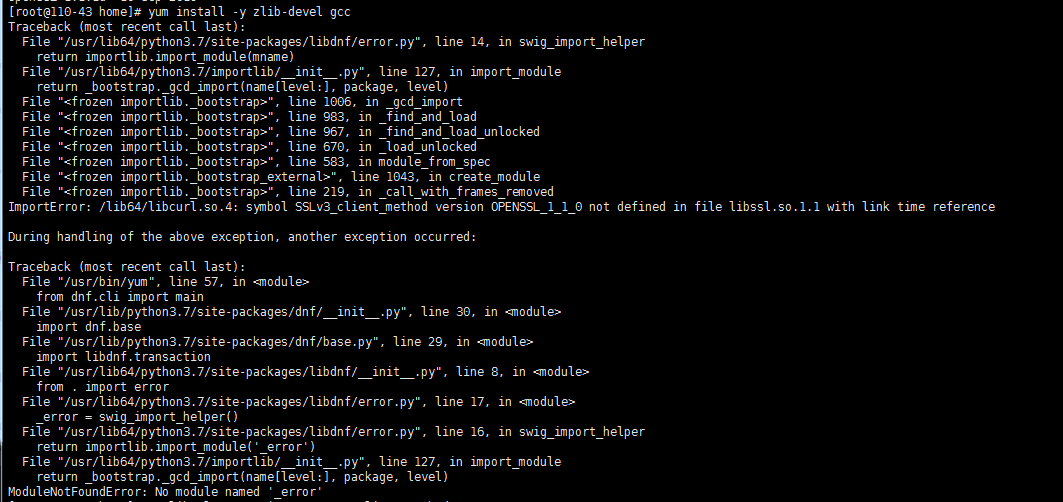
docker stop milvus-standalone

cd /home/yunqu/data/milvus

rm -rf volumes

## 升级openssl后yum无法使用

ImportError: /lib64/libcurl.so.4: symbol SSLv3\_ client. method version OPENSSL 1 1 0 not defined in file libssl.so.1.1 with link time reference



解决方式：

vi bin/yum

将 #!/usr/bin/python修改为 #!/usr/bin/python2.7 ，如下图所示：

文本

描述已自动生成

## 修改向量数据库端口

如现场需要修改向量数据库端口，请修改以下文件中的端口后，再重启服务：

vi /home/yunqu/data/vectordb/app/plugins/configs.py



## 使用docker-compose创建容器失败

如果出现*TypeError: load\_config() got an unexpected keyword argument 'config\_dict’*：

方法一：安装docker compose v2

方法二：将docker-compose添加到环境变量中，将docker-compose的解压路径加到/etc/profile里面。

tar -zxvf docker-compose.1.29.2.x86\_64.tgz -C /home/yunqu/bin/

vi /etc/profile

export PATH=/home/yunqu/bin:$PATH

source /etc/profile