summary: Install Milvus from Source Code

id: install-milvus-from-source-code

categories: milvus

tags: build

status: Published

authors: Brother Long

Feedback Link: https

author: @Zilliz.com

date: 5/20/2022

titlepage: true

titlepage-color: 175FFF

titlepage-text-color: FFFFFF

titlepage-rule-color: FFFFFF

title: Install Milvus from Source Code

Install Milvus from Source Code

Requirements

- Operating system
    - Ubuntu 18.04 or higher
    - CentOS 7
- GCC 7.0 or higher to support C++ 17
- CMake 3.14 or higher
- Git

For GPU-enabled version, you will also need:

- CUDA 10.0 or higher
- NVIDIA driver 418 or higher

Download milvus source from GitHub

```
git clone https://github.com/milvus-io/milvus
```

Install dependencies

Install in Ubuntu

```
$ cd [Milvus root path]/core
$ ./ubuntu_build_deps.sh
```

Install in CentOS

```
$ cd [Milvus root path]/core
$ ./centos7_build_deps.sh
```

Build from source

```
$ cd [Milvus root path]/core
$ ./build.sh -t Debug
```

or

```
$ ./build.sh -t Release
```

By default, it will build CPU-only version. To build GPU version, add `-g` option.

```
$ ./build.sh -g
```

If you want to know the complete build options, run the following command.

```
$./build.sh -h
```

When the build is completed, everything that you need in order to run Milvus will be installed under `[Milvus root path]/core/milvus`.

Launch Milvus server

```
$ cd [Milvus root path]/core/milvus
```

Add `lib/` directory to `LD_LIBRARY_PATH`

```
$ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:[Milvus root path]/core/milvus/lib
```

Then start Milvus server:

```
$ cd scripts
$ ./start_server.sh
```

To stop Milvus server, run:

```
$ ./stop_server.sh
```

Troubleshooting

Error message: `protocol https not supported or disabled in libcurl` Follow the steps below to solve this problem:

1. Make sure you have `libcurl4-openssl-dev` installed in your system.
2. Try reinstalling the latest CMake from source with `--system-curl` option:

```
$ ./bootstrap --system-curl
$ make
$ sudo make install
```

If the `--system-curl` command doesn't work, you can also reinstall CMake in **Ubuntu Software** on your local computer.

Error message: `internal compiler error` Try increasing the memory allocated to Docker. If this doesn't work, you can reduce the number of threads in CMake build in `[Milvus root path]/core/build.sh`.

```
make -j 8 install || exit 1 # The default number of threads is 8.
```

Note: You might also need to configure CMake build for faiss in `[Milvus root path]/core/src/index/thirdparty/faiss`.

Error message: `error while loading shared libraries: libmysqlpp.so.3` Follow the steps below to solve this problem:

1. Check whether `libmysqlpp.so.3` is correctly installed.
2. If `libmysqlpp.so.3` is installed, check whether it is added to `LD_LIBRARY_PATH`.

CMake version is not supported Follow the steps below to install a supported version of CMake:

1. Remove the unsupported version of CMake.
2. Get CMake 3.14 or higher. Here we get CMake 3.14.

```
$ wget https://cmake.org/files/v3.14/cmake-3.14.7-Linux-x86_64.tar.gz
```

3. Extract the file and install CMake.

```
$ tar zxvf cmake-3.14.7-Linux-x86_64.tar.gz
$ mv cmake-3.14.7-Linux-x86_64 /opt/cmake-3.14.7
$ ln -sf /opt/cmake-3.14.7/bin/* /usr/bin/
```