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

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

Install dependencies

Install in Ubuntu

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

Install in CentOS

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

Build from source

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

or

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

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

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

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

```
1  $./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

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

Add `lib/` directory to `LD_LIBRARY_PATH`

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

Then start Milvus server:

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

To stop Milvus server, run:

```
1  $ ./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:

```
1  $ ./bootstrap --system-curl
2  $ make
3  $ 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`.

```
1  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.

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

3. Extract the file and install CMake.

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