Ubuntu 安装和配置mongodb开发环境

操作系统：Ubuntu14.04

安装mongocxx驱动以及服务器配置，以下是安装步骤。

参考文档：

|  |
| --- |
| https://mongodb.github.io/mongo-cxx-driver/mongocxx-v3/installation/  [csdn博客: linux 下安装mongocxx] https://blog.csdn.net/qq\_35001989/article/details/76703824  [csdn博客: CentOS7下mongoc安装，验证] https://blog.csdn.net/qq\_35001989/article/details/76652949  [csdn博客：cmake 安装] https://blog.csdn.net/qq\_35001989/article/details/76618795 |

# 一．安装编译工具

|  |
| --- |
| sudo apt-get install -y gcc g++ git build-essential libtool automake autoconf wget curl |

# 二．安装cmake3.9

## 1. 编译安装

|  |
| --- |
| wget https://cmake.org/files/v3.9/cmake-3.9.0.tar.gz  tar -xvf cmake-3.9.0.tar.gz  cd cmake-3.9.0  ./configure  make -j4  sudo make install |

## 2. 测试验证

|  |
| --- |
| mkdir -p cmake\_test  cd cmake\_test  vim main.c  /\* ------------------- main.c : begin ------------------- \*/  #include <stdio.h>  int main()  {  printf("cmake test OK!\n");  return 0;  }  /\* ------------------- main.c : end ------------------- \*/    vim CMakeLists.txt (文件名不能改成其他)  /\* ------------------- CMakeLists.txt : begin ------------------- \*/  PROJECT(HELLO)  SET(SRC\_LIST main.c)  MESSAGE(STATUS "this is BINARY dir" ${HELLO\_BINARY\_DIR})  MESSAGE(STATUS "this is SOURCE dir" ${HELLO\_SOURCE\_DIR})  ADD\_EXECUTABLE(hello ${SRC\_LIST})  /\* ------------------- CMakeLists.txt : end ------------------- \*/  cmake . //"."代表当前目录  make  ./hello //输出cmake test OK!，到此完成cmake3.9.0安装验证 |

# 三．安装mongodb服务器

## 1. 安装

|  |
| --- |
| curl -O https://fastdl.mongodb.org/linux/mongodb-linux-x86\_64-3.0.6.tgz  tar -xvf mongodb-linux-x86\_64-3.0.6.tgz  sudo mv mongodb-linux-x86\_64-3.0.6 /usr/local/mongodb |

## 2. 配置

|  |
| --- |
| cd $HOME //进入home目录  mkdir -p mongo\_data  cd mongo\_data  mkdir -p logs  mkdir -p db  vim mongodb.conf  /\* -------------------- mongodb.conf : begin -------------------- \*/  dbpath=/home/dzq/mongo\_data/db  logpath=/home/dzq/mongo\_data/logs/arbiter.log  logappend=true  port=28000  bind\_ip=127.0.0.1,192.168.27.150  fork=true  journal=true  /\* -------------------- mongodb.conf : end -------------------- \*/  /\* 启动服务器 \*/  /usr/local/mongodb/bin/mongod -f /home/dzq/mongo\_data/mongodb.conf  /\* 启动shell客户端 \*/  /usr/local/mongodb/bin/mongo 127.0.0.1:28000 |

# 四．安装mongoc驱动

## 1. 编译安装

|  |
| --- |
| wget https://github.com/MongoDB/mongo-c-driver/releases/download/1.9.4/mongo-c-driver-1.9.4.tar.gz  tar -xvf mongo-c-driver-1.9.4.tar.gz  cd mongo-c-driver-1.9.4  ./configure --disable-automatic-init-and-cleanup  make -j4  sudo make install |

## 2. 添加环境变量

|  |
| --- |
| vim $HOME/.bashrc  /\* ------------------- .bashrc : begin ------------------- \*/  PATH=$PATH:$HOME/bin:/usr/local/mongodb/bin  export PATH  LD\_LIBRARY\_PATH=/usr/local/lib  export LD\_LIBRARY\_PATH  CPATH=$CPATH:/usr/local/include/libmongoc-1.0:/usr/local/include/libbson-1.0  export CPATH  /\* ------------------- .bashrc : end ------------------- \*/  /\* 使.bashrc立即生效 \*/  source $HOME/.bashrc |

## 3. 测试验证

|  |
| --- |
| mkdir -p mongoc\_test  cd mongoc\_test  vim connect.c  /\* ------------------- connect.c : begin ------------------- \*/  #include "bson.h"  #include "bcon.h"  #include "mongoc.h"  int main (int argc, char \*argv[])  {  mongoc\_client\_t \*client;  mongoc\_database\_t \*database;  mongoc\_collection\_t \*collection;  bson\_t \*command, reply, \*insert;  bson\_error\_t error;  char \*str;  bool retval;  mongoc\_init ();  client = mongoc\_client\_new ("mongodb://localhost:28000");  mongoc\_client\_set\_appname (client, "connect-example");  database = mongoc\_client\_get\_database (client, "db\_name");  collection = mongoc\_client\_get\_collection (client, "db\_name", "coll\_name");  command = BCON\_NEW ("ping", BCON\_INT32 (1));  retval = mongoc\_client\_command\_simple (client, "admin", command, NULL, &reply, &error);  if (!retval) {  fprintf (stderr, "%s\n", error.message);  return EXIT\_FAILURE;  }  str = bson\_as\_json (&reply, NULL);  printf ("%s\n", str);  insert = BCON\_NEW ("hello", BCON\_UTF8 ("world"));  if (!mongoc\_collection\_insert (collection, MONGOC\_INSERT\_NONE, insert, NULL, &error)) {  fprintf (stderr, "%s\n", error.message);  }  bson\_destroy (insert);  bson\_destroy (&reply);  bson\_destroy (command);  bson\_free (str);  mongoc\_collection\_destroy (collection);  mongoc\_database\_destroy (database);  mongoc\_client\_destroy (client);  mongoc\_cleanup ();  return 0;  }  /\* ------------------- connect.c : end ------------------- \*/  gcc -o connect connect.c -lmongoc-1.0 -lbson-1.0  ./connect //如果配置成功则在终端输出 { "ok" : 1.0 } |

# 五．安装mongocxx驱动

## 1. 编译安装

|  |
| --- |
| curl -OL https://github.com/mongodb/mongo-cxx-driver/archive/r3.2.0.tar.gz  tar -xvf r3.2.0.tar.gz  cd mongo-cxx-driver-r3.2.0/build  cmake -DCMAKE\_BUILD\_TYPE=Release -DBSONCXX\_POLY\_USE\_MNMLSTC=1 -DCMAKE\_INSTALL\_PREFIX=/usr/local/mongo-cxx-driver-r3.2.0 -DLIBBSON\_DIR=/usr/local/lib -DLIBMONGOC\_DIR=/usr/local/lib ..  sudo make EP\_mnmlstc\_core  make -j4  sudo make install |

## 2. 添加环境变量

|  |
| --- |
| vim $HOME/.bashrc  /\* ------------------- .bashrc : begin ------------------- \*/  CPATH=$CPATH:/usr/local/mongo-cxx-driver-r3.2.0/include/bsoncxx/v\_noabi:/usr/local/mongo-cxx-driver-r3.2.0/include/mongocxx/v\_noabi  export CPATH  /\* ------------------- .bashrc : end ------------------- \*/  /\* 使.bashrc立即生效 \*/  source $HOME/.bashrc  /\* 添加动态链接库的路径 \*/  sudo ldconfig /usr/local/mongo-cxx-driver-r3.2.0/lib |

## 3. 测试验证

|  |
| --- |
| mkdir -p mongocxx\_test  cd mongocxx\_test  vim test.cpp  /\* --------------------- test.cpp : begin --------------------- \*/  #include <iostream>  #include <bsoncxx/builder/stream/document.hpp>  #include <bsoncxx/json.hpp>  #include <mongocxx/client.hpp>  #include <mongocxx/instance.hpp>  int main(int argc, char \*argv[])  {  mongocxx::instance inst{};  mongocxx::client conn{mongocxx::uri("mongodb://localhost:28000")};  bsoncxx::builder::stream::document document{};  auto collection = conn["testdb"]["testcollection"];  document << "hello" << "world";  collection.insert\_one(document.view());  auto cursor = collection.find({});  for (auto&& doc : cursor) {  std::cout << bsoncxx::to\_json(doc) << std::endl;  }  }  /\* --------------------- test.cpp : end --------------------- \*/    g++ --std=c++11 test.cpp -o test -L /usr/local/mongo-cxx-driver-r3.2.0/lib -lmongocxx -lbsoncxx  ./test  正确安装验证会显示如下：  { "\_id" : { "$oid" : "5adb3a621d41c858be0a2382" }, "hello" : "world" } |