环境我还没搭建完，需要安装的包我都装了。现在出问题看不出哪的原因

# 初始化环境

IOS：ubuntu-16.04.3-server-amd64.iso

IP：192.168.40.111

虚拟机上网方式：NAT

|  |
| --- |
| root@ubuntu:~# lsb\_release -a  No LSB modules are available.  Distributor ID: Ubuntu  Description: Ubuntu 16.04.3 LTS  Release: 16.04  Codename: xenial |

# 安装build-essential

|  |
| --- |
| apt-get install build-essential |

# 安装wget

|  |
| --- |
| apt-get install wget |

# 安装[log4cpp](http://log4cpp.sourceforge.net/)

下载最新log4cpp包，目前最新版本log4cpp-1.1.3

|  |
| --- |
| wget https://nchc.dl.sourceforge.net/project/log4cpp/log4cpp-1.1.x%20%28new%29/log4cpp-1.1/log4cpp-1.1.3.tar.gz |

解压log4cpp-1.1.3.tar.gz

|  |
| --- |
| tar -zxf log4cpp-1.1.3.tar.gz |

从0.2.0开始，log4cpp支持autoconf,依次执行。log4cpp装在了/usr下面



|  |
| --- |
| ./configure --prefix=/usr/log4cpp --with-pthreads |
| make |
| make check |
| make iinstall |

# 安装[libcurl](https://curl.haxx.se/libcurl/)

下载最新log4cpp包，目前最新版本curl-7.61.1.tar.gz

|  |
| --- |
| wget https://curl.haxx.se/download/curl-7.61.1.tar.gz |

解压curl-7.61.1.tar.gz

|  |
| --- |
| tar -zxf curl-7.61.1.tar.gz |

安装指导<https://curl.haxx.se/docs/install.html，libcurl>安装在了/usr下面



|  |
| --- |
| ./configure --prefix=/usr/curl --enable-pthreads |
| make |
| make check |
| make iinstall |

# 安装[jsoncpp](https://json.org/)

参考：<https://launchpad.net/ubuntu/xenial/+package/libjsoncpp-dev>

jsoncpp包是和我的Linux版本对应的

下载jsoncpp包

wget <https://launchpadlibrarian.net/250172610/libjsoncpp_1.7.2.orig.tar.gz>

安装jsoncpp步骤

安装jsoncpp前需要线安装scons

wget <https://nchc.dl.sourceforge.net/project/scons/scons/2.4.1/scons-2.4.1.tar.gz>

tar zxf scons-2.4.1.tar.gz

python setup.py install

scons安装目录

|  |
| --- |
| Installed SCons library modules into /usr/local/lib/scons-2.4.1  Installed SCons scripts into /usr/local/bin  Installed SCons man pages into /usr/local/man/man1 |

设置环境变量

export MYSCONS=/download/scons-2.4.1

export SCONS\_LIB\_DIR=$MYSCONS/engine

解压libjsoncpp\_1.7.2.orig.tar.gz，进入jsoncpp-1.7.2执行

scons platform=linux-gcc

生成的静态库，动态库在libs下边，我还没动

生成的头文件在include下面，我还没动

# 安装[tinyxml](http://www.grinninglizard.com/tinyxml/)

\*Please Note\* that TinyXML development has stopped and all development has moved to TinyXML-2.

下载最新tinyxml包，目前最新版本tinyxml\_2\_6\_2.zip

参考文章：[Linux 下配置Tinyxml,将其编译为静态库](https://blog.csdn.net/zyx_linux/article/details/21948523)

[Linux 下配置Tinyxml,将其编译为动态库](https://blog.csdn.net/zyx_linux/article/details/21957501)

下载tinyxml\_2\_6\_2.zip

wget https://launchpadlibrarian.net/290269646/tinyxml\_2.6.2.orig.tar.gz

# 安装libevent

下载wget https://launchpadlibrarian.net/310424743/libevent\_2.0.21-stable.orig.tar.gz

解压libevent\_2.0.21-stable.orig.tar.gz

tar zxf libevent\_2.0.21-stable.orig.tar.gz

cd libevent-2.0.21-stable/

./configure --prefix=/usr/libevent

make

make verify可选，这步比较耗时

make install

# 安装lua

https://launchpad.net/ubuntu/+source/lua5.2/5.2.4-1ubuntu1

下载lua5.2

wget <https://launchpadlibrarian.net/253731968/lua5.2_5.2.4.orig.tar.gz>

tar zxf lua5.2\_5.2.4.orig.tar.gz

cd lua5.2-5.2.4

make posix

make posix install

自己指定安装目录的话使用下面命令

make local

make install INSTALL\_TOP=../../../usr/lua5.2

使用lua5.3的下载

wget <http://archive.ubuntu.com/ubuntu/pool/main/l/lua5.3/lua5.3_5.3.1.orig.tar.gz>

安装过程同lua5.2

# 安装MySQL

安装制定版本mysql参考：<https://blog.csdn.net/u011060906/article/details/79612408>

我是直接安装最新版的

apt-get install mysql-server

apt install mysql-client

apt install libmysqlclient-dev

按转包完毕版本

root@ubuntu:~# mysql --version

mysql Ver 14.14 Distrib 5.7.23, for Linux (x86\_64) using EditLine wrapper

# 安装thrift

参考：<https://thrift-tutorial.readthedocs.io/en/latest/installation.html>

先安装依赖

apt-get install libboost-dev libboost-test-dev libboost-program-options-dev libboost-filesystem-dev libboost-thread-dev libevent-dev automake libtool flex bison pkg-config g++ libssl-dev

我们使用的语言是C++，所以需要先安装boost。

参考：<http://thrift.apache.org/docs/install/>

下载boost

wget <https://nchc.dl.sourceforge.net/project/boost/boost/1.53.0/boost_1_53_0.tar.gz>

tar zxf boost\_1\_53\_0.tar.gz

cd boost\_1\_53\_0/

boost安装参考：<https://blog.csdn.net/fall221/article/details/9090939>

./bootstrap.sh

./bjam

下载thrift

wget <http://mirrors.shu.edu.cn/apache/thrift/0.11.0/thrift-0.11.0.tar.gz>

tar zxf thrift-0.11.0.tar.gz

cd thrift-0.11.0/

./bootstrap.sh

export LUA=/usr/lua5.3/bin/lua

export LUA\_INCLUDE=-I/usr/lua5.3/include

export LUA\_LIB=-llua

./configure --with-cpp --with-lua=no

make

make install