NETWORKING & SYSTEM ADMINISTRATION LAB

Name: Justin v kalappura

Roll No: 10

Batch: MCA -B

Date: 06-06-2022

Experiment No.: 28

Aim:

Build and install software from source code, familiarity with make and cmake utilities expected.

Procedure:

1. We can obtain information of a package and its dependencies using the apt command. Doing that for cmake:

   $ apt show cmake

```
mca@S66:~/Documents/neha/CMake$ apt show cmake
Package: cmake
Version: 3.10.2-1ubuntu2
Priority: optional
Section: devel
Origin: Ubuntu
Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Original-Maintainer: Debian CMake Team <pkg-cmake-team@lists.alioth.debian.org>
Bugs: https://bugs.launchpad.net/ubuntu/+filebug
Installed-Size: 17.3 MB
Depends: cmake-data (= 3.10.2-1ubuntu2), procps, libarchive13 (>= 3.0.4), libc6 (>= 2.15), libcurl4 (>= 7.16.2), libexpat1 (>= 2.0.1), libgcc1
 (>= 1:3.0), libjsoncpp1 (>= 1.7.4), librhash0 (>= 1.2.6), libstdc++6 (>= 5.2), libuv1 (>= 1.4.2), zlib1g (>= 1:1.2.3.3)
Recommends: gcc, make
Suggests: cmake-doc, ninja-build
Homepage: https://cmake.org/
Supported: 5y
Download-Size: 3,138 kB
APT-Sources: http://in.archive.ubuntu.com/ubuntu bionic/main amd64 Packages
Description: cross-platform, open-source make system
 CMake is used to control the software compilation process using
 simple platform and compiler independent configuration files. CMake
 generates native makefiles and workspaces that can be used in the
 compiler environment of your choice. CMake is quite sophisticated: it
 is possible to support complex environments requiring system
 configuration, pre-processor generation, code generation, and template
 instantiation.
 .
 CMake was developed by Kitware as part of the NLM Insight
 Segmentation and Registration Toolkit project. The ASCI VIEWS project
 also provided support in the context of their parallel computation
 environment. Other sponsors include the Insight, VTK, and VXL open
 source software communities.
```

2. To install cmake , g++ and make using the apt command, type:
   $ sudo apt install cmake g++ make

```
mca@S66:~/Documents/neha/CMake$ sudo apt install cmake g++ make
[sudo] password for mca:
Reading package lists... Done
Building dependency tree
Reading state information... Done
g++ is already the newest version (4:7.3.0-3ubuntu2).
make is already the newest version (4.1-9.1ubuntu1).
make set to manually installed.
The following additional packages will be installed:
  cmake-data libcurl4 libjsoncpp1 librhash0 libuv1
Suggested packages:
  cmake-doc ninja-build
The following NEW packages will be installed:
  cmake cmake-data libcurl4 libjsoncpp1 librhash0 libuv1
0 upgraded, 6 newly installed, 0 to remove and 1 not upgraded.
3 not fully installed or removed.
Need to get 4,900 kB of archives.
After this operation, 25.3 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu bionic/main amd64 cmake-data all 3.10.2-1ubuntu2 [1,331 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu bionic/main amd64 libcurl4 amd64 7.58.0-2ubuntu3 [214 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu bionic/main amd64 libjsoncpp1 amd64 1.7.4-3 [73.6 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu bionic/main amd64 librhash0 amd64 1.3.6-2 [78.1 kB]
Get:5 http://in.archive.ubuntu.com/ubuntu bionic/main amd64 libuv1 amd64 1.18.0-3 [64.4 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu bionic/main amd64 cmake amd64 3.10.2-1ubuntu2 [3,138 kB]
Fetched 4,900 kB in 1s (6,846 kB/s)
Selecting previously unselected package cmake-data.
(Reading database ... 171165 files and directories currently installed.)
Preparing to unpack .../0-cmake-data_3.10.2-1ubuntu2_all.deb ...
Unpacking cmake-data (3.10.2-1ubuntu2) ...
Selecting previously unselected package libcurl4:amd64.
Preparing to unpack .../1-libcurl4_7.58.0-2ubuntu3_amd64.deb ...
Unpacking libcurl4:amd64 (7.58.0-2ubuntu3) ...
Selecting previously unselected package libjsoncpp1:amd64.
Preparing to unpack .../2-libjsoncpp1_1.7.4-3_amd64.deb ...
Unpacking libjsoncpp1:amd64 (1.7.4-3) ...
```

A Sample CMake project:

```
mca@S66:~/Documents/neha/CMake$ mkdir Myproject
mca@S66:~/Documents/neha/CMake$ cd Myproject
```

```
mca@S66:~/Documents/neha/CMake/Myproject$ gedit hello_world.cpp
```

```
#include <iostream>
        int main() {
        std::cout<<"Hello World!"<<std::endl;
        return 0 ;
        }
```

```
mca@S66:~/Documents/neha/CMake/Myproject$ gedit hello_world.cpp
```

```
cmake_minimum_required(VERSION 3.10)
project(MyProject)
add_executable(hello hello_world.cpp)
```

A directory to which CMake was executed is called “Built Directory”

```
mca@S66:~/Documents/neha/CMake/Myproject$ mkdir build
mca@S66:~/Documents/neha/CMake/Myproject$ cd build
```

```
mca@S66:~/Documents/neha/CMake/Myproject/build$ cmake ..
-- The C compiler identification is GNU 7.3.0
-- The CXX compiler identification is GNU 7.3.0
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Configuring done
-- Generating done
-- Build files have been written to: /home/mca/Documents/neha/CMake/Myproject/build
```

Text Editor

```
mca@S66:~/Documents/neha/CMake/Myproject/build$ cmake --build .
Scanning dependencies of target hello
[ 50%] Building CXX object CMakeFiles/hello.dir/hello_world.cpp.o
[100%] Linking CXX executable hello
```

```
mca@S66:~/Documents/neha/CMake/Myproject/build$ ./hello
Hello World!
mca@S66:~/Documents/neha/CMake/Myproject/build$
```