Windows下使用MinGW+Cmake编译n2n

记录下windows下MinGW+cmake的使用方法，以Win7 64位下编译n2n为例。

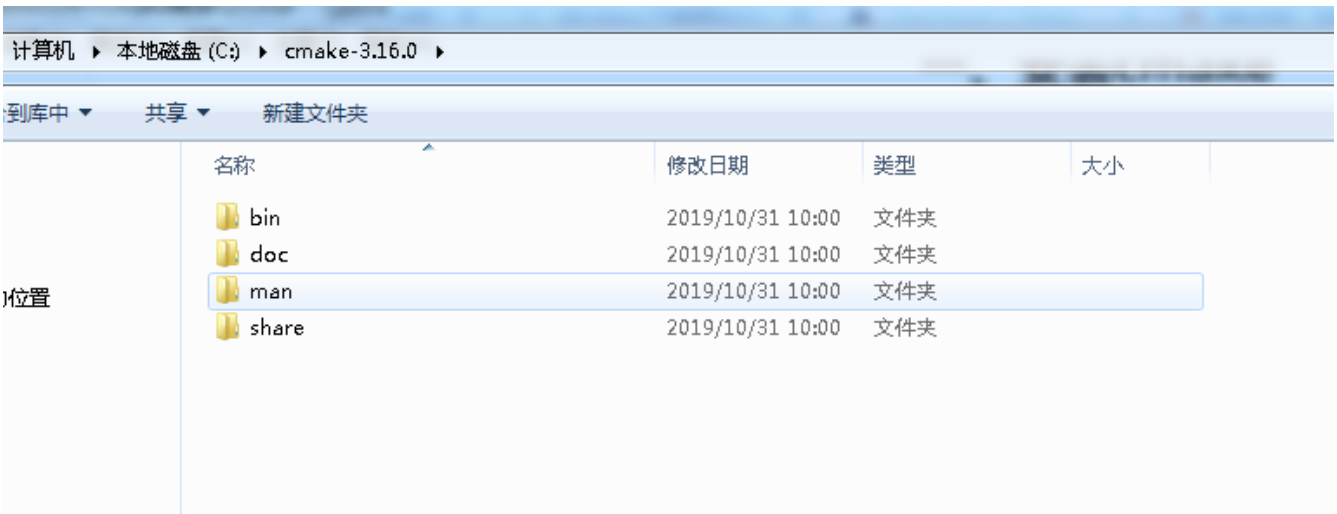
一、安装cmake

1.下载并解压

https://cmake.org/download/，以压缩包版为例

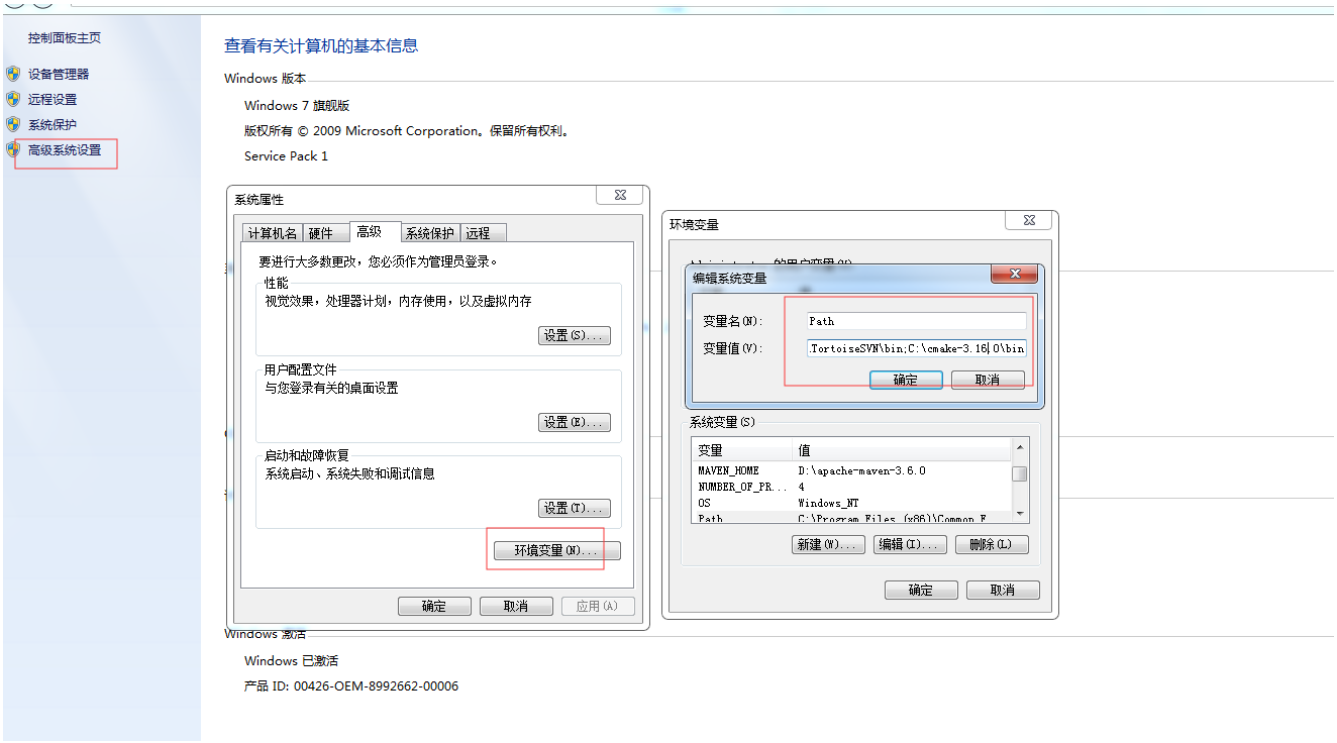
Platform	Files
Windows win64-x64 Installer: **Installer tool has changed. Uninstall CMake 3.4 or lower first!**	cmake-3.16.0-rc3-win64-x64.msi
Windows win64-x64 ZIP	cmake-3.16.0-rc3-win64-x64.zip
Windows win32-x86 Installer: **Installer tool has changed. Uninstall CMake 3.4 or lower first!**	cmake-3.16.0-rc3-win32-x86.msi
Windows win32-x86 ZIP	cmake-3.16.0-rc3-win32-x86.zip
Mac OS X 10.7 or later	cmake-3.16.0-rc3-Darwin-x86_64.dmg
	cmake-3.16.0-rc3-Darwin-x86_64.tar.gz
Linux x86_64	cmake-3.16.0-rc3-Linux-x86_64.sh
	cmake-3.16.0-rc3-Linux-x86_64.tar.gz

解压到任意目录，如解压到C盘根目录：

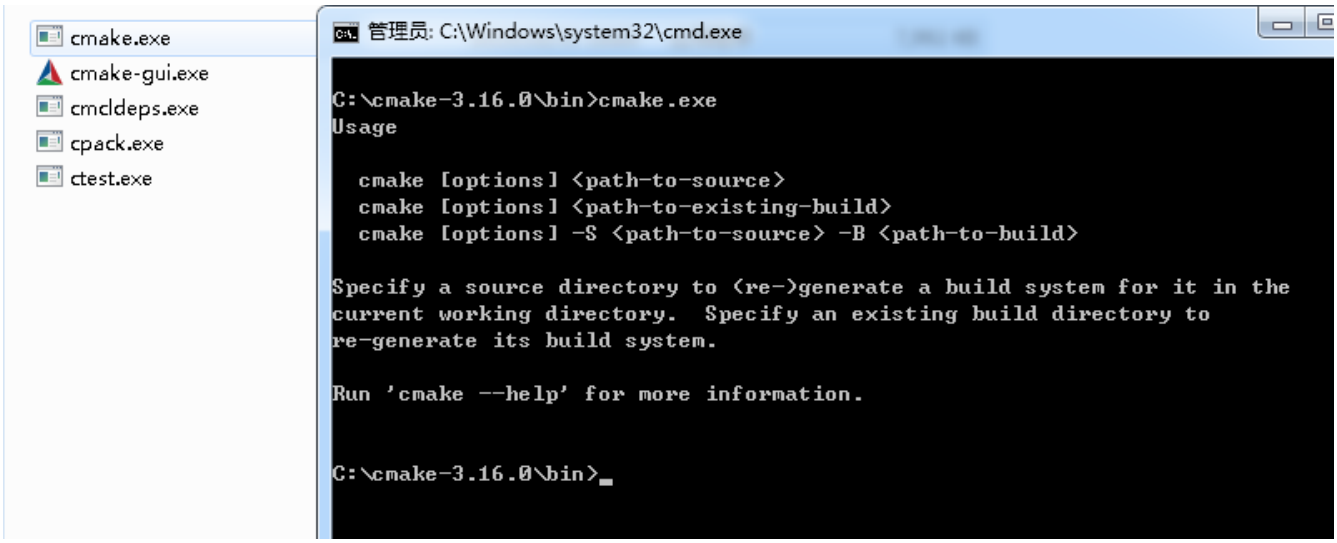


2.设置环境变量

在环境变量Path的尾部增加 **;C:\cmake-3.12.1-win32-x86\bin 注意路径和前面的分号**



重启，打开CMD，输入 cmake，检查是否链接成功

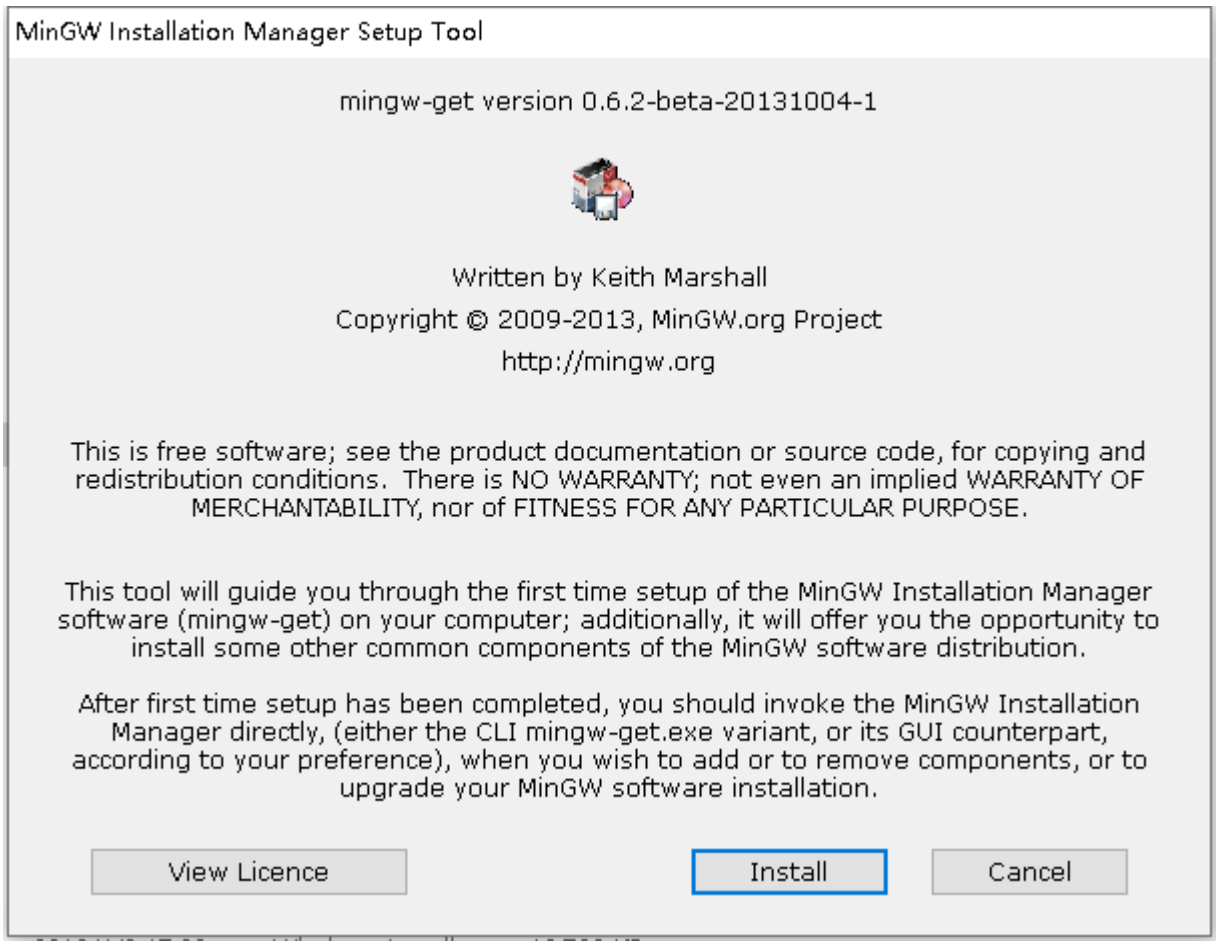


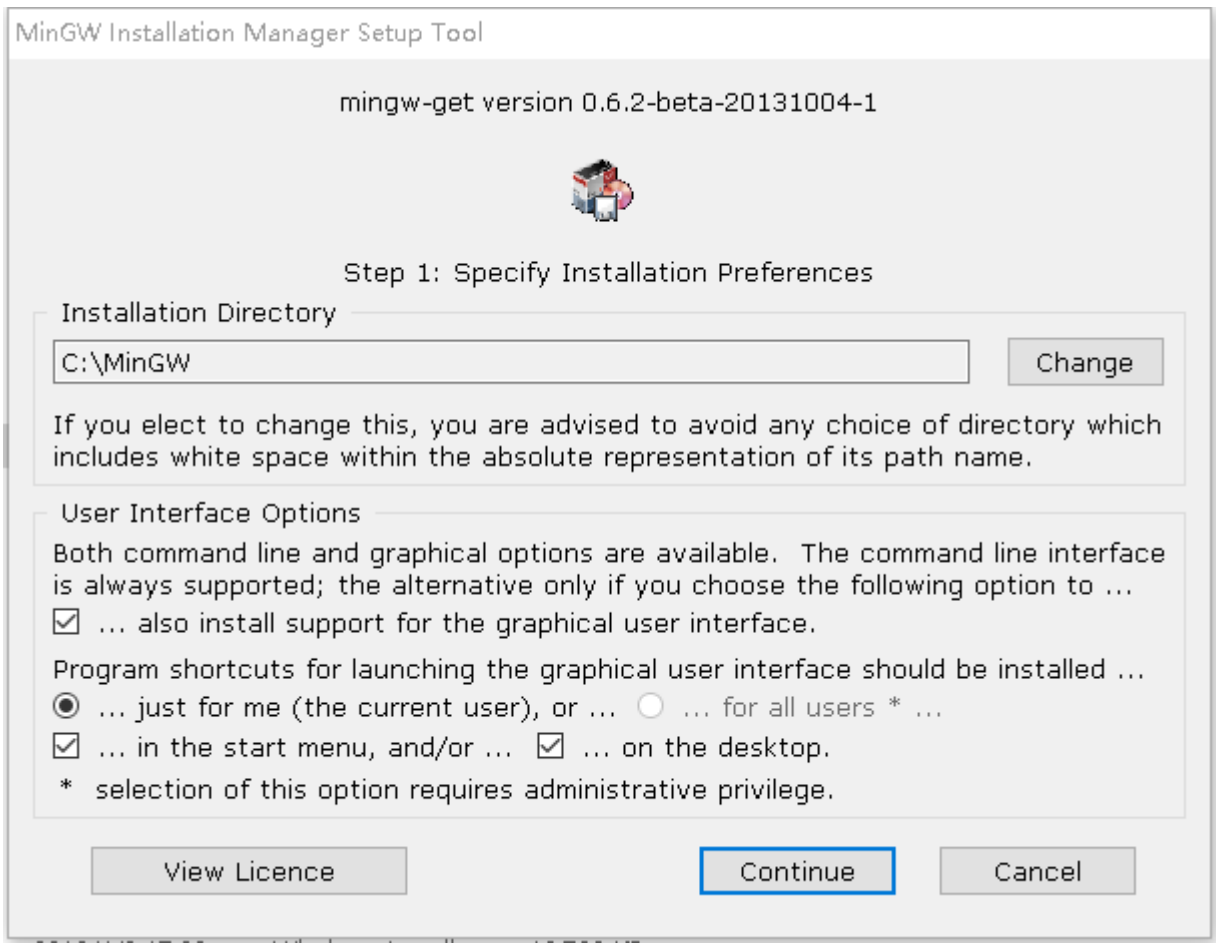
二、安装MinGW

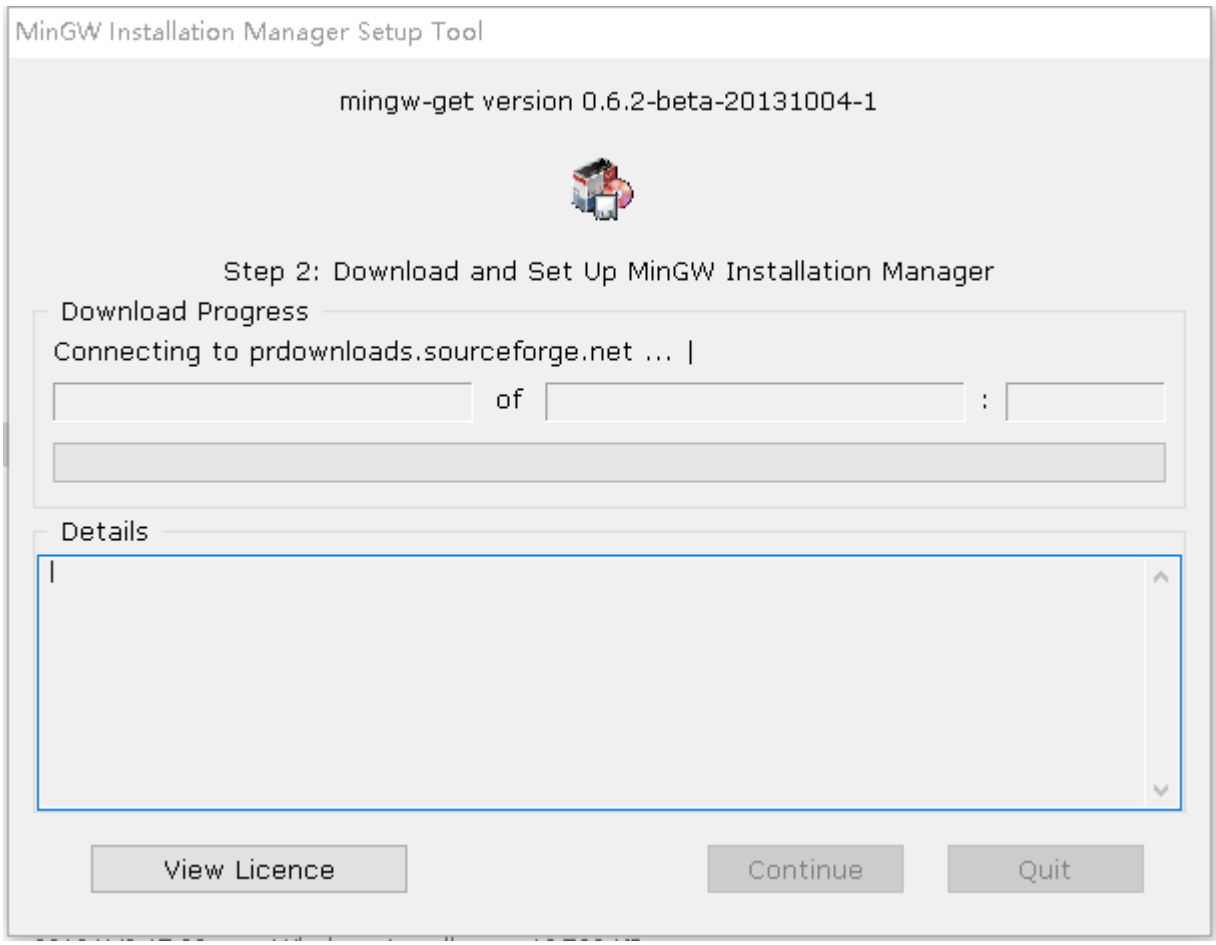
1.下载安装器

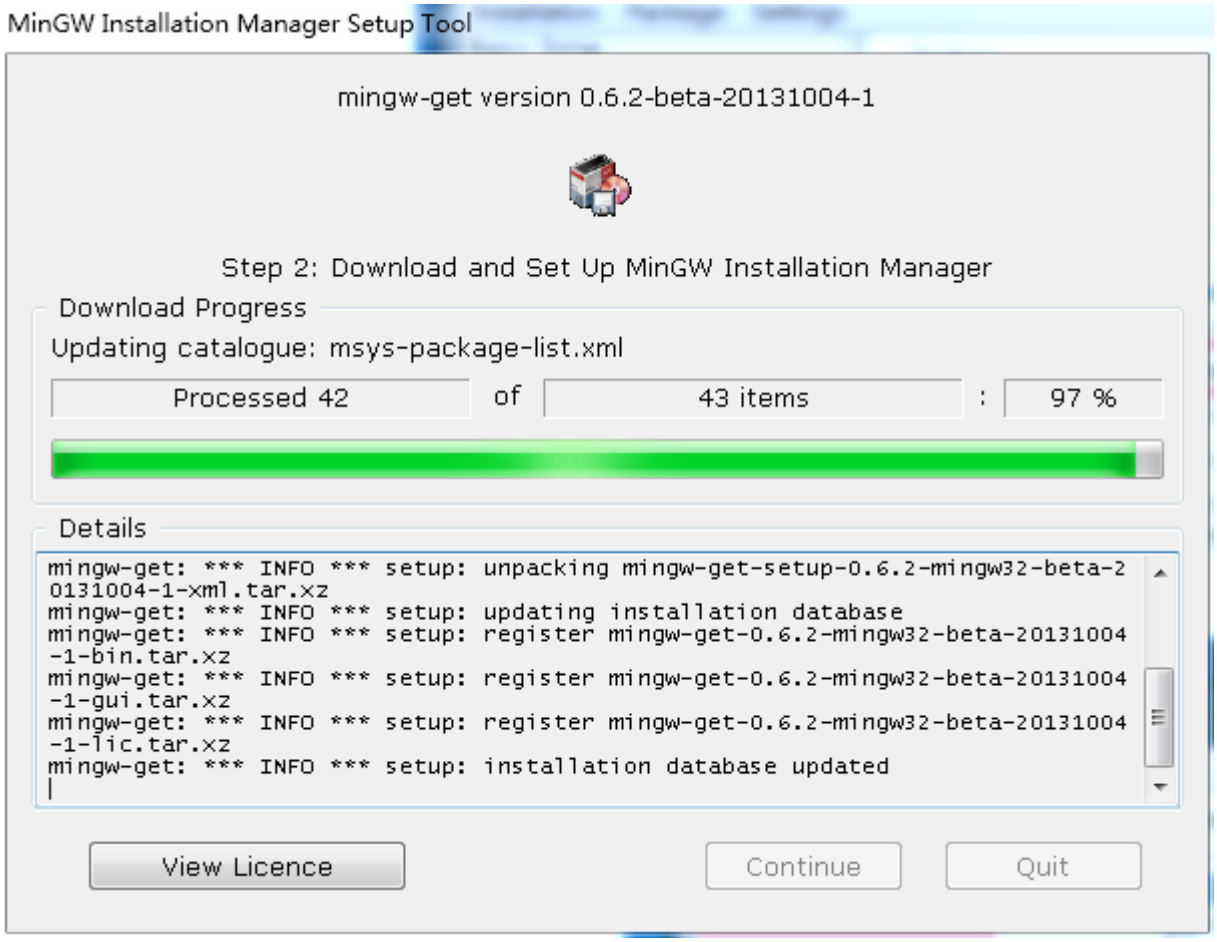
https://sourceforge.net/projects/mingw/files/latest/download

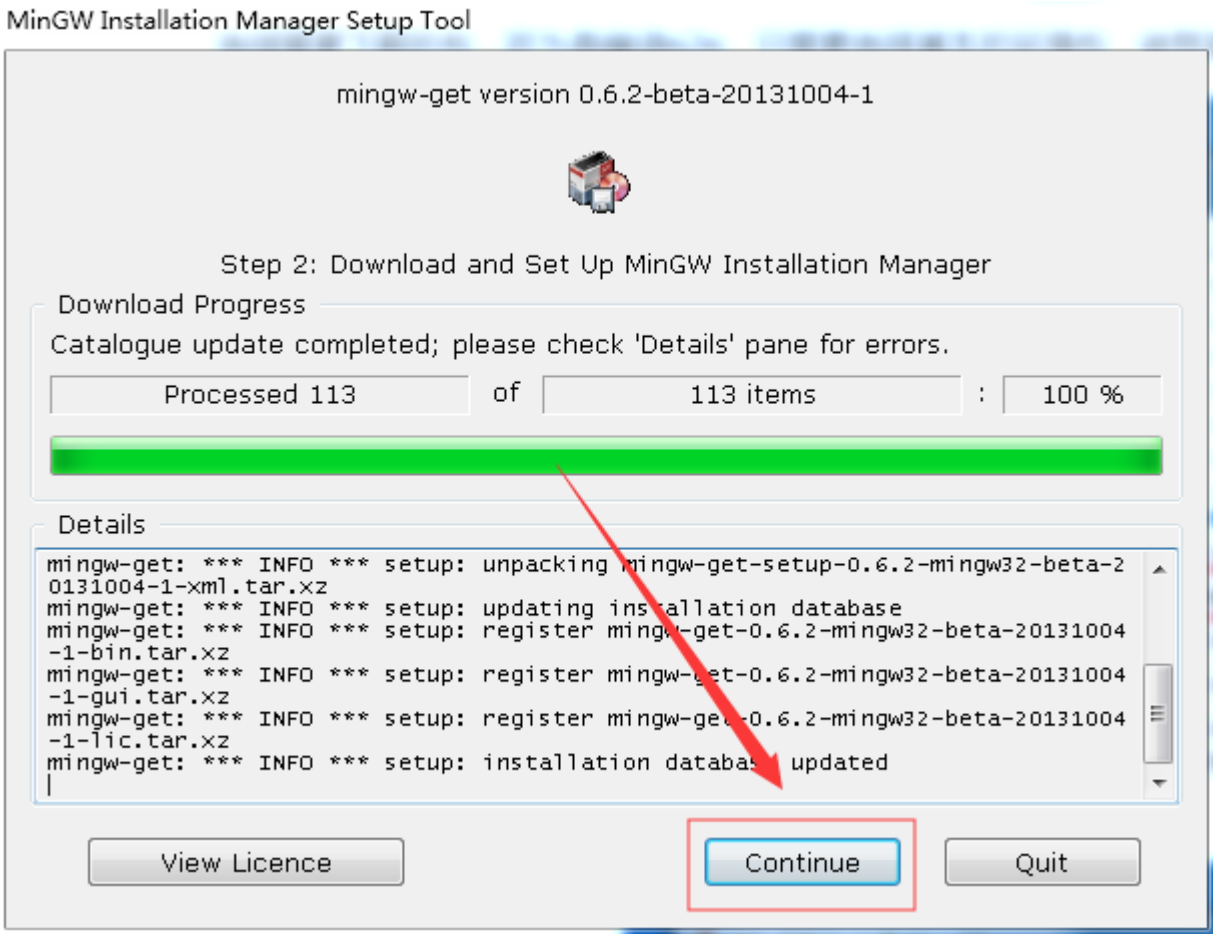
运行并指定安装目录后，便会开始在线下载安装器的文件，耐心等待


MinGW Installation Manager Setup Tool
mingw-get version 0.6.2-beta-20131004-1
Written by Keith Marshall
Copyright © 2009-2013, MinGW.org Project
http://mingw.org
This is free software; see the product documentation or source code, for copying and redistribution conditions. There is NO WARRANTY; not even an implied WARRANTY OF MERCHANTABILITY, nor of FITNESS FOR ANY PARTICULAR PURPOSE.
This tool will guide you through the first time setup of the MinGW Installation Manager software (mingw-get) on your computer; additionally, it will offer you the opportunity to install some other common components of the MinGW software distribution.
After first time setup has been completed, you should invoke the MinGW Installation Manager directly, (either the CLI mingw-get.exe variant, or its GUI counterpart, according to your preference), when you wish to add or to remove components, or to upgrade your MinGW software installation.
View Licence
Install
Cancel

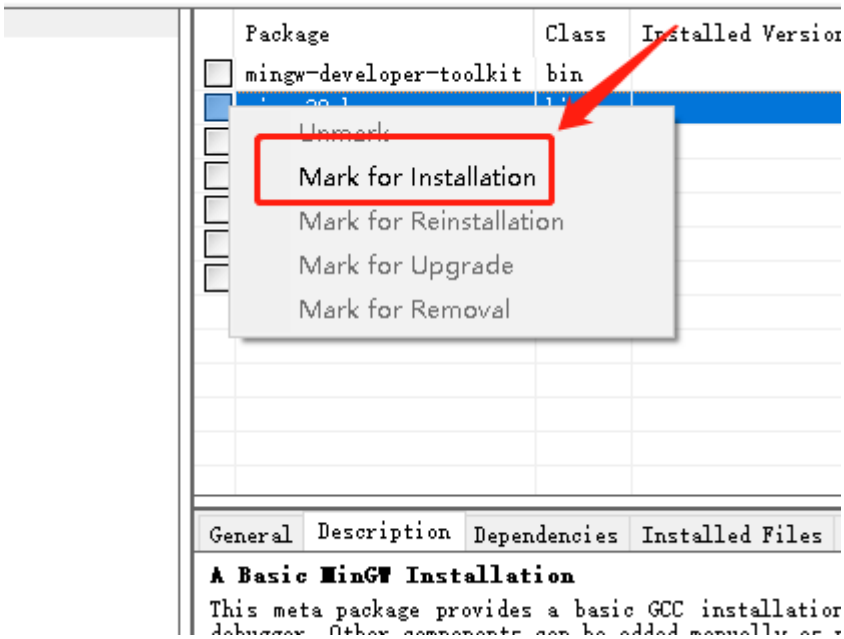

MinGW Installation Manager Setup Tool
mingw-get version 0.6.2-beta-20131004-1
Step 1: Specify Installation Preferences
Installation Directory
C:\MinGW
Change
If you elect to change this, you are advised to avoid any choice of directory which includes white space within the absolute representation of its path name.
User Interface Options
Both command line and graphical options are available. The command line interface is always supported; the alternative only if you choose the following option to ...
... also install support for the graphical user interface.
Program shortcuts for launching the graphical user interface should be installed ...
... just for me (the current user), or ... ... for all users * ...
... in the start menu, and/or ... ... on the desktop.
* selection of this option requires administrative privilege.
View Licence
Continue
Cancel

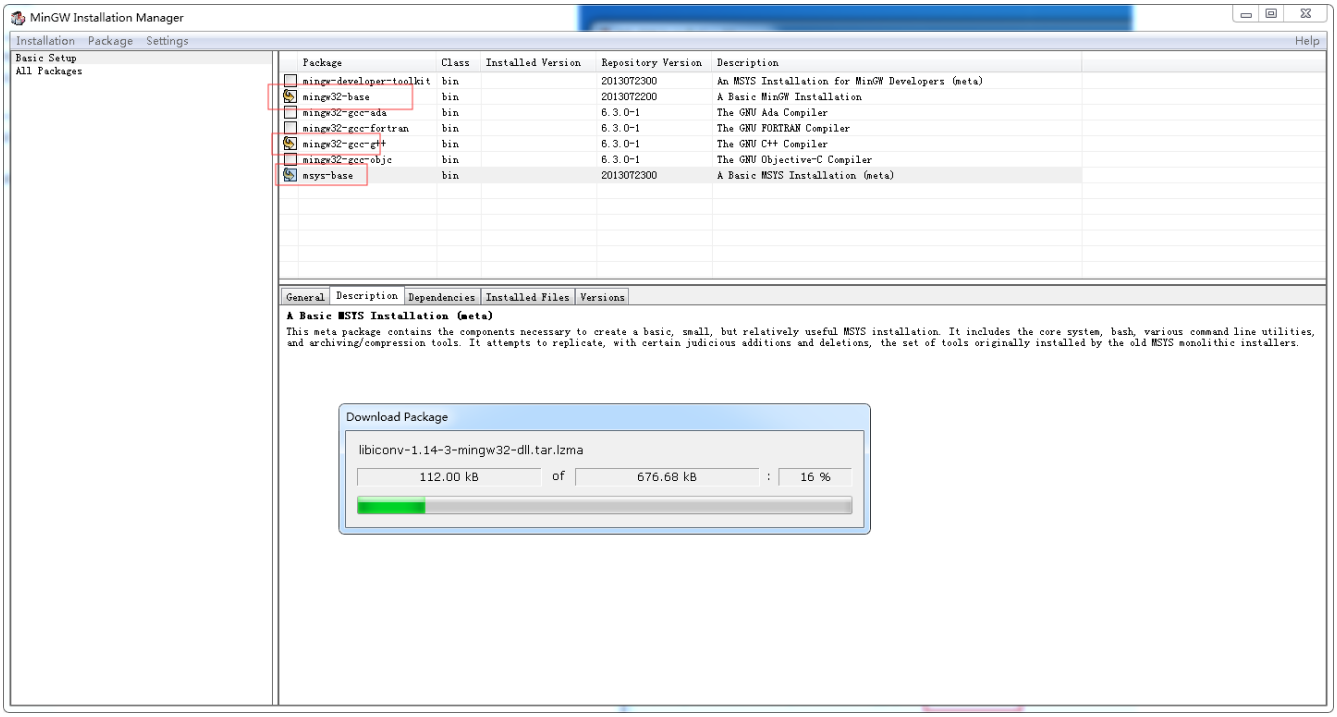

MinGW Installation Manager Setup Tool
mingw-get version 0.6.2-beta-20131004-1
Step 2: Download and Set Up MinGW Installation Manager
Download Progress
Connecting to prdownloads.sourceforge.net ... |
of
:
Details
View Licence
Continue
Quit

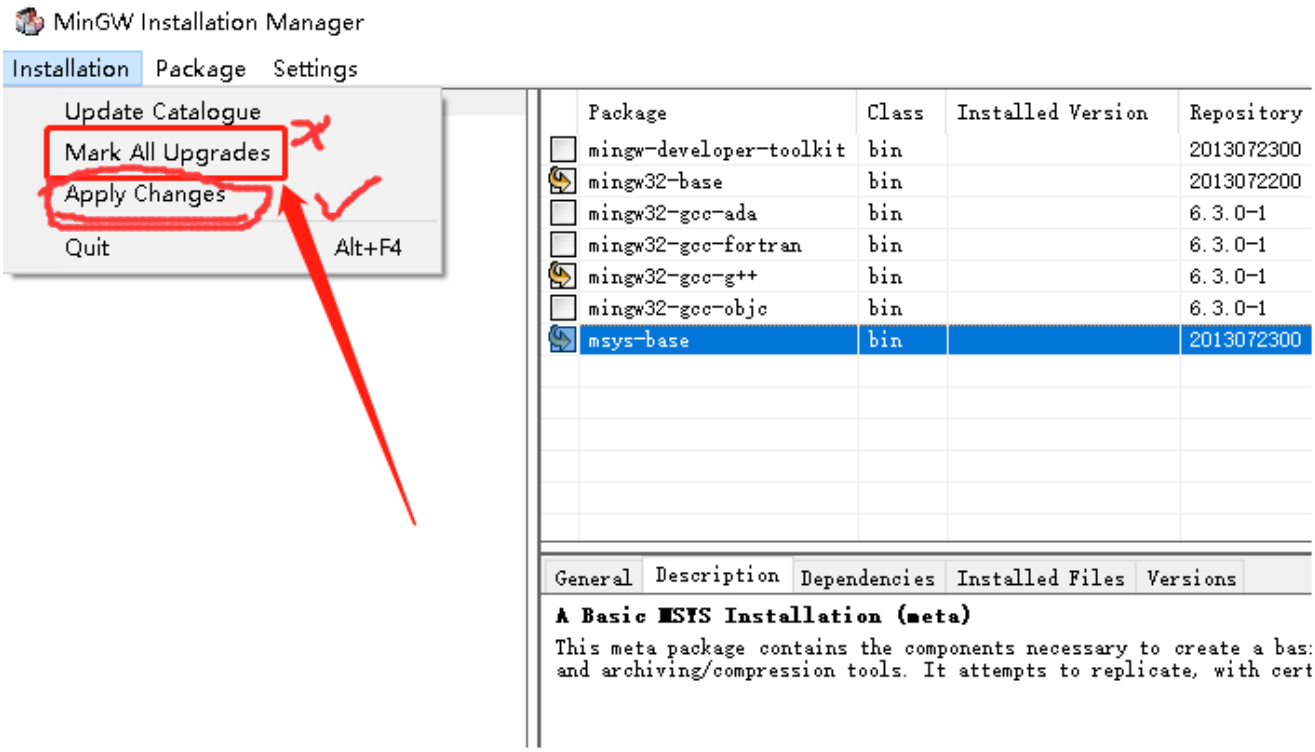

MinGW Installation Manager Setup Tool
mingw-get version 0.6.2-beta-20131004-1
Step 2: Download and Set Up MinGW Installation Manager
Download Progress
Updating catalogue: msys-package-list.xml
Processed 42
of
43 items
:
97 %
Details
mingw-get: *** INFO *** setup: unpacking mingw-get-setup-0.6.2-mingw32-beta-20131004-1-xml.tar.xz
mingw-get: *** INFO *** setup: updating installation database
mingw-get: *** INFO *** setup: register mingw-get-0.6.2-mingw32-beta-20131004-1-bin.tar.xz
mingw-get: *** INFO *** setup: register mingw-get-0.6.2-mingw32-beta-20131004-1-gui.tar.xz
mingw-get: *** INFO *** setup: register mingw-get-0.6.2-mingw32-beta-20131004-1-lic.tar.xz
mingw-get: *** INFO *** setup: installation database updated
View Licence
Continue
Quit

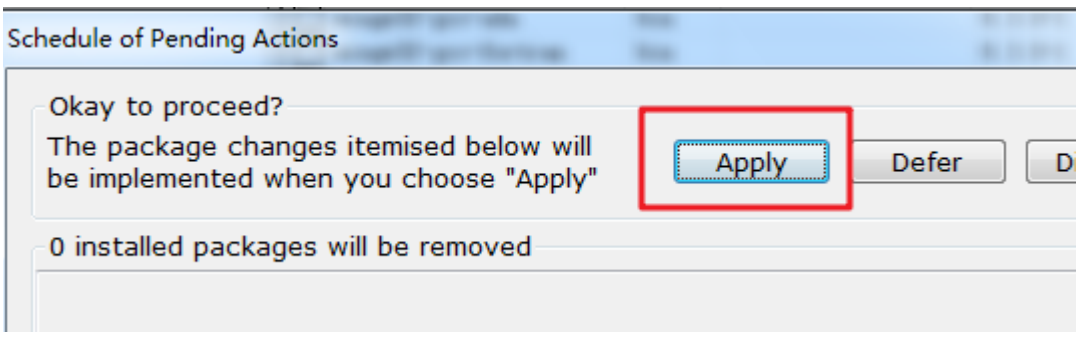


选择需要下载的包，因为是编译n2n，只需要选择基本的环境包，依然是耐心等待



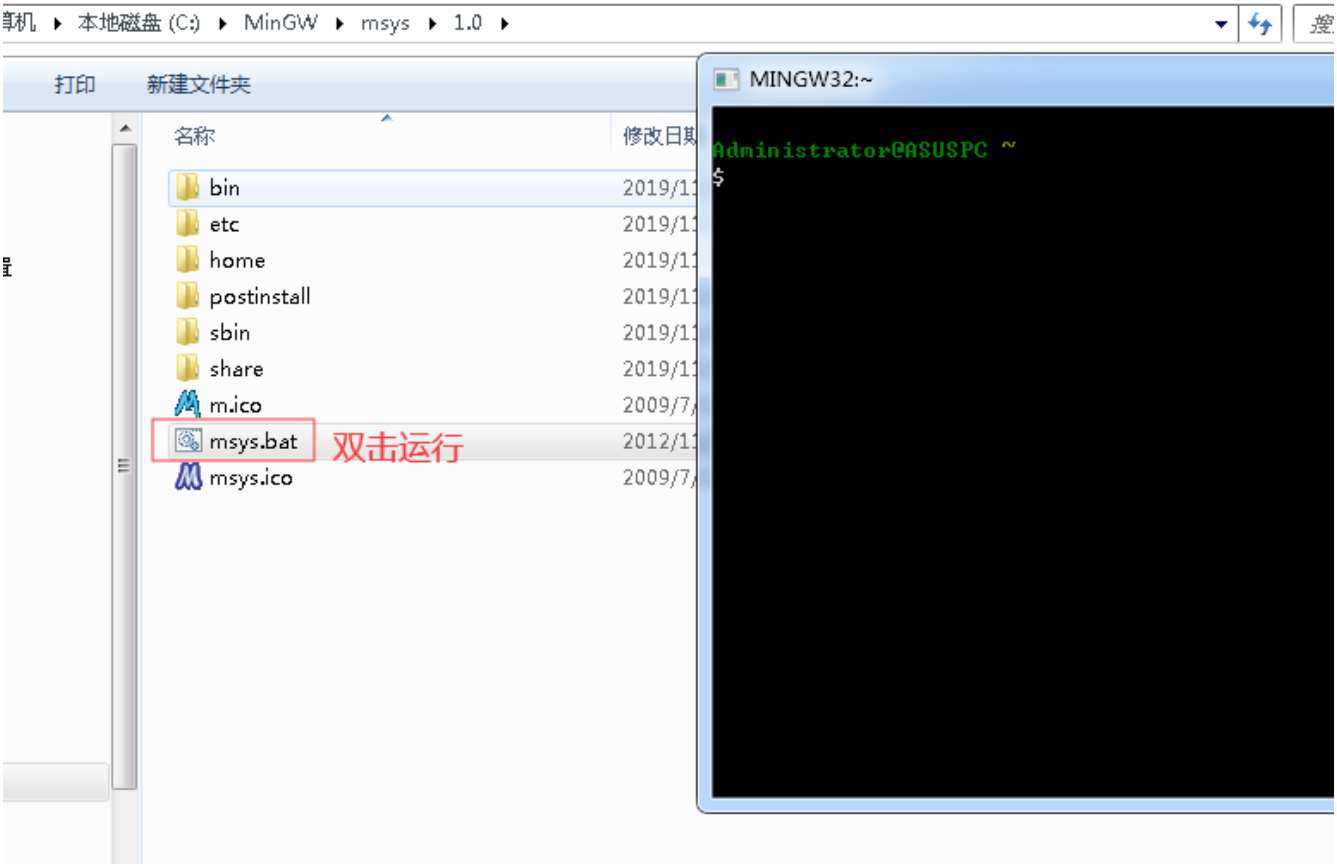

MinGW Installation Manager
Installation Package Settings
Help
Basic Setup
All Packages
Package Class Installed Version Repository Version Description
mingw-developer-toolkit bin 2013072300 An MSYS Installation for MinGW Developers (meta)
mingw32-base bin 2013072200 A Basic MinGW Installation
mingw32-gcc-ada bin 6.3.0-1 The GNU Ada Compiler
mingw32-gcc-fortran bin 6.3.0-1 The GNU FORTRAN Compiler
mingw32-gcc-g++ bin 6.3.0-1 The GNU C++ Compiler
mingw32-gcc-objc bin 6.3.0-1 The GNU Objective-C Compiler
msys-base bin 2013072300 A Basic MSYS Installation (meta)
General Description Dependencies Installed Files Versions
A Basic MSYS Installation (meta)
This meta package contains the components necessary to create a basic, small, but relatively useful MSYS installation. It includes the core system, bash, various command line utilities, and archiving/compression tools. It attempts to replicate, with certain judicious additions and deletions, the set of tools originally installed by the old MSYS monolithic installers.
Download Package
libiconv-1.14-3-mingw32-dll.tar.lzma
112.00 kB of 676.68 kB : 16 %


MinGW Installation Manager
Installation Package Settings
Update Catalogue
Mark All Upgrades
Apply Changes
Quit Alt+F4
Package Class Installed Version Repository
mingw-developer-toolkit bin 2013072300
mingw32-base bin 2013072200
mingw32-gcc-ada bin 6.3.0-1
mingw32-gcc-fortran bin 6.3.0-1
mingw32-gcc-g++ bin 6.3.0-1
mingw32-gcc-objc bin 6.3.0-1
msys-base bin 2013072300
General Description Dependencies Installed Files Versions
A Basic MSYS Installation (meta)
This meta package contains the components necessary to create a bas:
and archiving/compression tools. It attempts to replicate, with cert


Schedule of Pending Actions
Okay to proceed?
The package changes itemised below will be implemented when you choose "Apply"
Apply
Defer
0 installed packages will be removed

2.测试

运行 C:\MinGW\msys\1.0\msys.bat 进入命令行窗口



三、开始编译

将下载的n2n源码放至任意文件夹，比如C盘根目D:\soft\n2n\n2n_v2

依次运行

```
cd D:\soft\n2n\n2n_v2
mkdir build
cd build
cmake -G "MSYS Makefiles" --build ./ ../
make
```

```
Administrator@ASUSPC /d/soft
$ cd n2n/

Administrator@ASUSPC /d/soft/n2n
$ ls
README.md  n2n_v1  n2n_v2

Administrator@ASUSPC /d/soft/n2n
$ cd n2n_v2/

Administrator@ASUSPC /d/soft/n2n/n2n_v2
$ ls
CMakeLists.txt         gen_keyfile.py  n2n_transforms.h  tuntap_freebsd.c
COPYING                lzoconf.h       n2n_v2.7          tuntap_linux.c
HACKING                lzodefs.h       n2n_wire.h        tuntap_netbsd.c
INSTALL                minilzo.c       openwrt           tuntap_osx.c
NEW_FEATURES.txt       minilzo.h       scm.h             twofish.c
README                 munin           scripts           twofish.h
benchmark.c            n2n.c           sglib.h           unix-scm.c
benchmark_hashtable.c  n2n.h           sn.c              version.c
cmake                  n2n.spec        supernode.1       win32
debian                 n2n_keyfile.c   transform_aes.c   wire.c
edge.8                 n2n_keyfile.h   transform_null.c  wireshark
edge.c                 n2n_test.c      transform_tf.c

Administrator@ASUSPC /d/soft/n2n/n2n_v2
$ mkdir build

Administrator@ASUSPC /d/soft/n2n/n2n_v2
$ cd build/

Administrator@ASUSPC /d/soft/n2n/n2n_v2/build
$ cmake -G "MSYS Makefiles" --build ./ ../
-- The C compiler identification is GNU 6.3.0
-- The CXX compiler identification is GNU 6.3.0
-- Check for working C compiler: C:/MinGW/bin/gcc.exe
-- Check for working C compiler: C:/MinGW/bin/gcc.exe -- 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: C:/MinGW/bin/g++.exe
-- Check for working CXX compiler: C:/MinGW/bin/g++.exe -- 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: D:/soft/n2n/n2n_v2/build
```

配置成功，执行 make 进行编译，出错，提示wintap.c 的 213行 ERROR_IO_PENDING 未定义

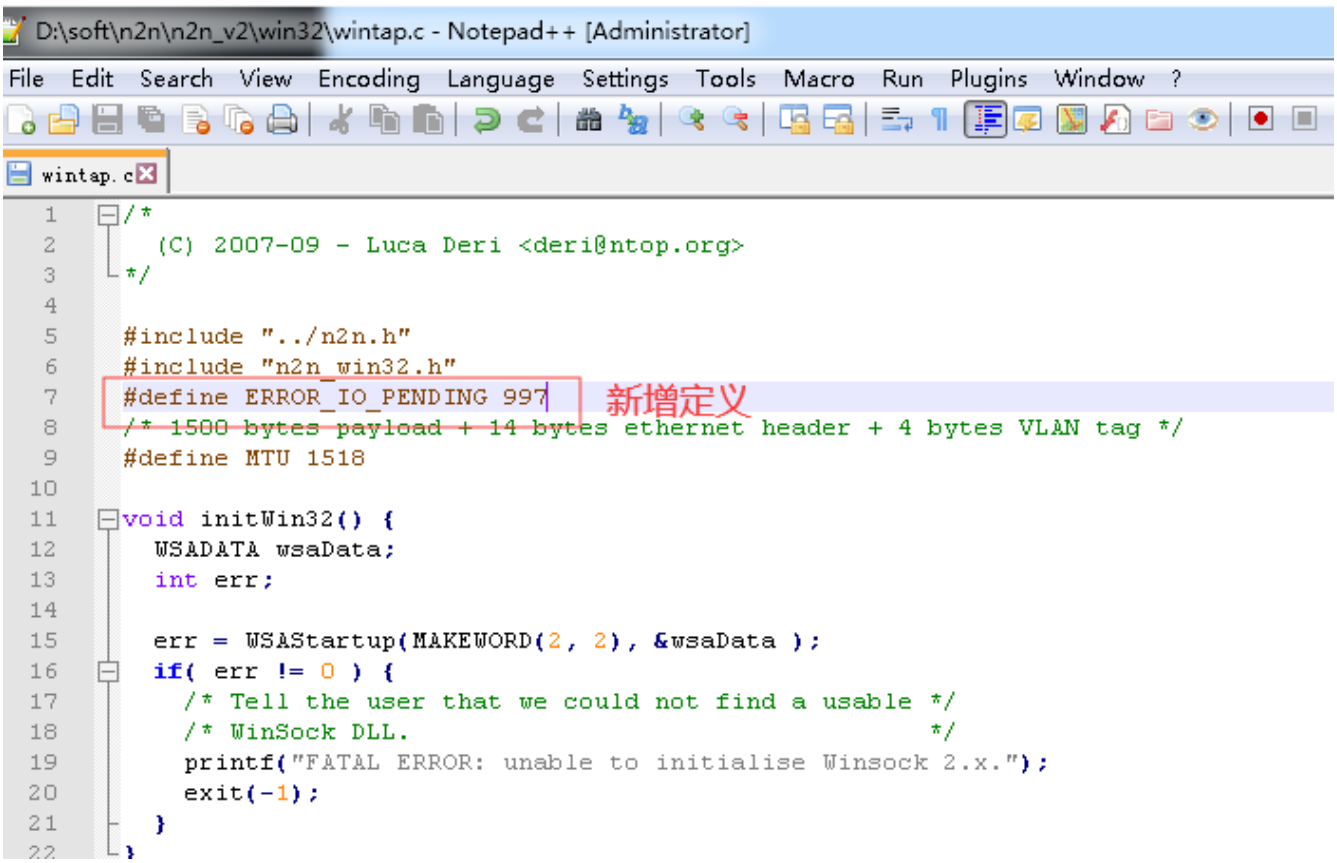
```
$ make
Scanning dependencies of target n2n_win32
[  3%] Building C object win32/CMakeFiles/n2n_win32.dir/getopt1.c.obj
[  6%] Building C object win32/CMakeFiles/n2n_win32.dir/getopt.c.obj
[  9%] Building C object win32/CMakeFiles/n2n_win32.dir/wintap.c.obj
d:/soft/n2n/n2n_v2/win32/wintap.c: In function 'tuntap_read':
d:/soft/n2n/n2n_v2/win32/wintap.c:213:8: error: 'ERROR_IO_PENDING' undeclared (
irst use in this function)
   case ERROR_IO_PENDING:
        ^~~~~~~~~~~~~~~~~

d:/soft/n2n/n2n_v2/win32/wintap.c:213:8: note: each undeclared identifier is re
orted only once for each function it appears in
d:/soft/n2n/n2n_v2/win32/wintap.c: In function 'tuntap_write':
d:/soft/n2n/n2n_v2/win32/wintap.c:243:8: error: 'ERROR_IO_PENDING' undeclared (
irst use in this function)
   case ERROR_IO_PENDING:
        ^~~~~~~~~~~~~~~~~

make[2]: *** [win32/CMakeFiles/n2n_win32.dir/wintap.c.obj] Error 1
make[1]: *** [win32/CMakeFiles/n2n_win32.dir/all] Error 2
make: *** [all] Error 2
```

修改win32目录下的wintap.c，头部加入定义

```
#define ERROR_IO_PENDING 997
```



再次执行 make 编译

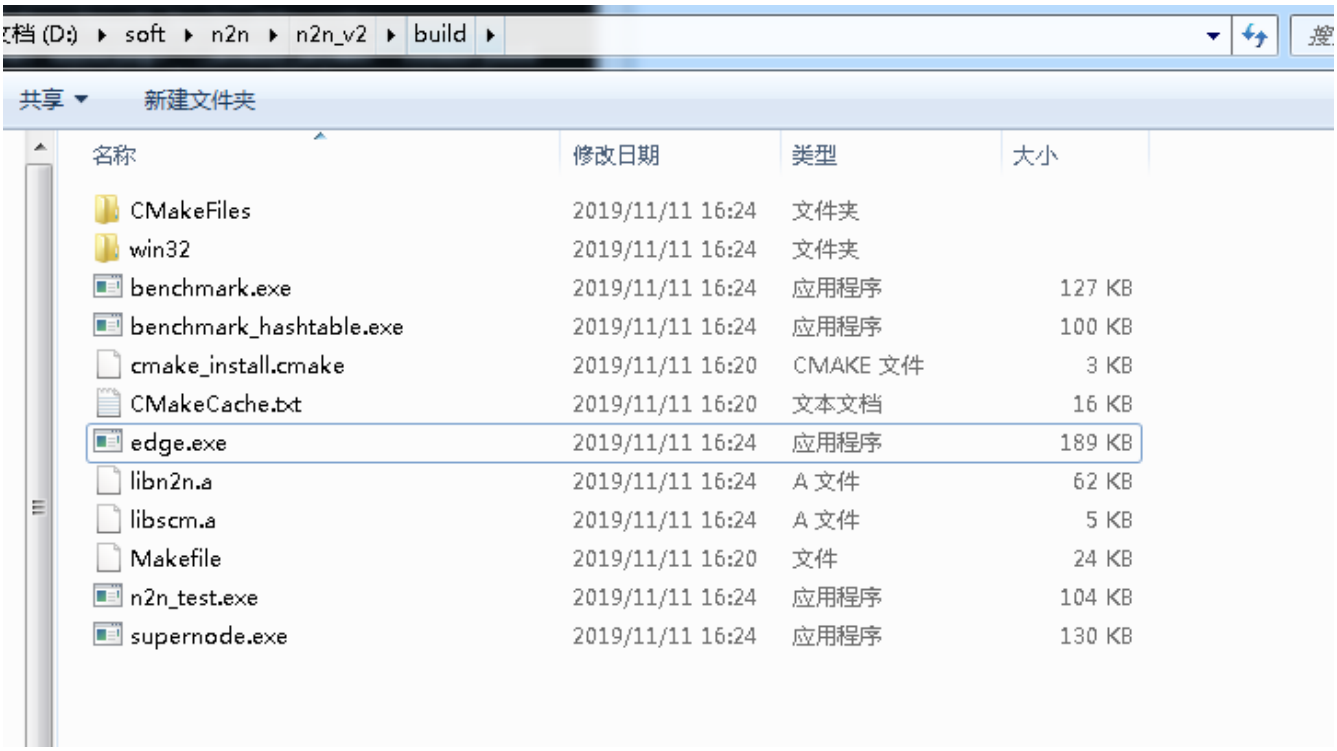
```
MINGW32:/d/soft/n2n/n2n_v2/build
[ 93%] Linking C executable n2n_test.exe
[ 93%] Built target n2n_test
Scanning dependencies of target benchmark_hashtable
[ 96%] Building C object CMakeFiles/benchmark_hashtable.dir/benchmark_hashtable.
c.obj
In file included from d:/soft/n2n/n2n_v2/n2n_wire.h:14:0,
                 from d:/soft/n2n/n2n_v2/benchmark_hashtable.c:1:
d:/soft/n2n/n2n_v2/win32/n2n_win32.h:16:0: warning: "_WIN32_WINNT" redefined
 #define _WIN32_WINNT 0x501 //Otherwise the linker doesnt find getaddrinfo

In file included from c:\mingw\include\w32api.h:59:0,
                 from c:\mingw\include\_mingw.h:73,
                 from c:\mingw\include\stdlib.h:49,
                 from d:/soft/n2n/n2n_v2/n2n_wire.h:11,
                 from d:/soft/n2n/n2n_v2/benchmark_hashtable.c:1:
c:\mingw\include\sdkddkver.h:185:0: note: this is the location of the previous d
efinition
 #  define _WIN32_WINNT _WIN32_WINNT_WIN2K

[100%] Linking C executable benchmark_hashtable.exe
[100%] Built target benchmark_hashtable

Administrator@ASUSPC /d/soft/n2n/n2n_v2/build
$
        半:
```

错误已解决，成功编译出可执行文件



```
D:\soft\n2n\n2n_v2\build>edge.exe -h
Welcome to n2n v.2.1.0 for Windows-6.1.7601
Built on Nov 11 2019 16:24:37
Copyright 2007-09 - http://www.ntop.org

edge -a [static:¦dhcp:]<tun IP address> -c <community> [-k <encrypt key> ¦ -K <key file>] [-s <netmask>] [-m <MAC address>]
-l <supernode host:port> [-p <local port>] [-M <mtu>] [-r] [-E] [-v] [-t <mgmt port>] [-b] [-h]

-a <mode:address>        ¦ Set interface address. For DHCP use '-r -a dhcp:0.0.0.0'
-c <community>           ¦ n2n community name the edge belongs to.
-k <encrypt key>         ¦ Encryption key (ASCII) - also N2N_KEY=<encrypt key>. Not with -K.
-K <key file>            ¦ Specify a key schedule file to load. Not with -k.
-s <netmask>             ¦ Edge interface netmask in dotted decimal notation (255.255.255.0).
-l <supernode host:port> ¦ Supernode IP:port
-L <local_ip>            ¦ Add local ip to bypass between same nat problem
-i <interval>            ¦ Set the NAT hole-punch interval (default 20seconds)
-b                       ¦ Periodically resolve supernode IP
                         : (when supernodes are running on dynamic IPs)
-p <local port>          ¦ Fixed local UDP port.
-m <MAC address>         ¦ Fix MAC address for the TAP interface (otherwise it may be random)
                         : eg. -m 01:02:03:04:05:06
-M <mtu>                 ¦ Specify n2n MTU of edge interface (default 1400).
-r                       ¦ Enable packet forwarding through n2n community.
-E                       ¦ Accept multicast MAC addresses (default=drop).
-v                       ¦ Make more verbose. Repeat as required.
-t                       ¦ Management UDP Port (for multiple edges on a machine).

Environment variables:
  N2N_KEY                ¦ Encryption key (ASCII). Not with -K or -k.
```