UAVSim Installation Manual

SERVER AND DESKTOP MODE UNIX INSTALLATIONS

Faraha Jahan and Ahmad Javaid

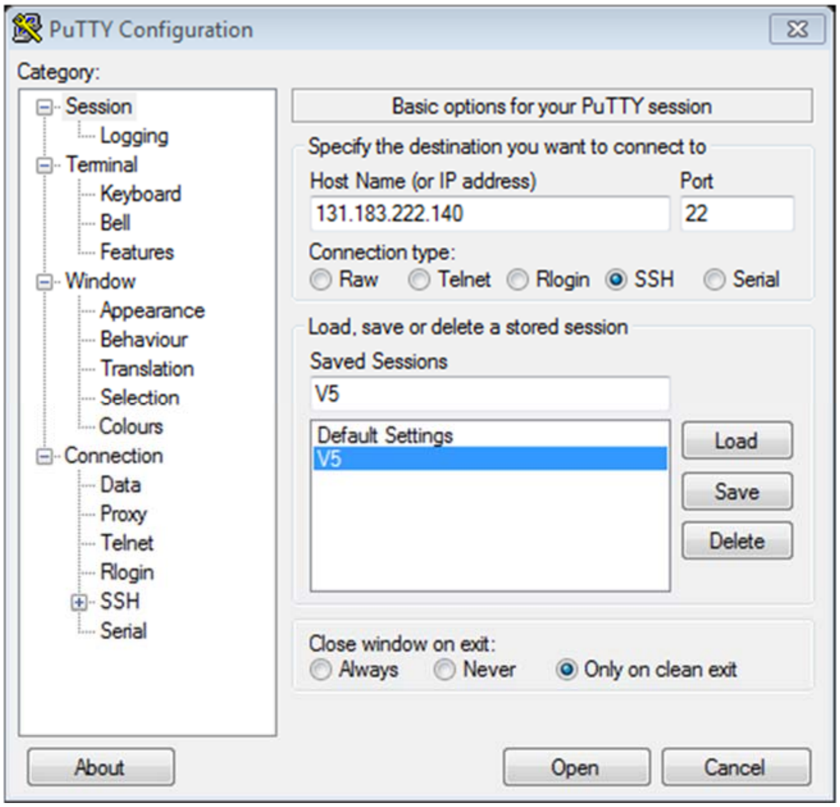
THE UNIVERSITY OF TOLEDO |

Table of Contents

A. PART 1: Installation on a Sever

1. Step 1: Connecting to the Sever

Connect to the machine you want to use as a server



Host Name (or IP address): 131.183.222.140

Port: 22

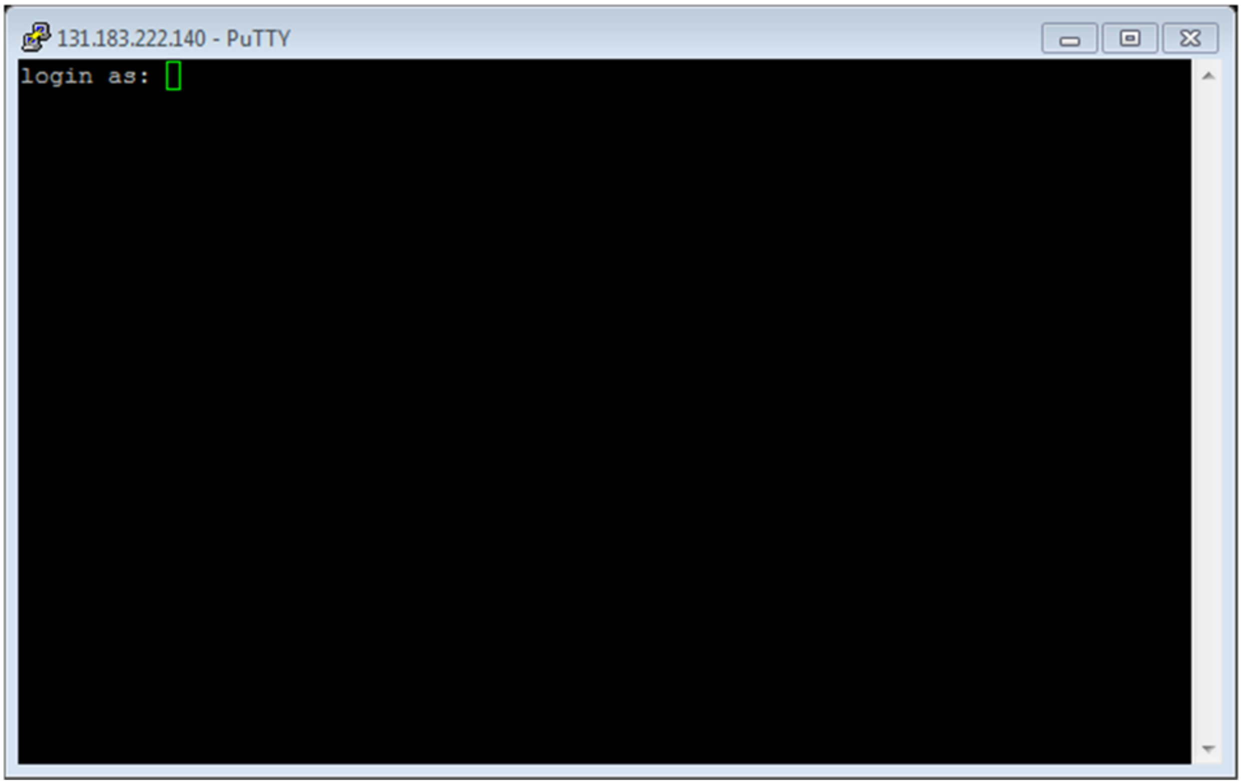
Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Saved Sessions: V5

Close window on exit: ○ Always ○ Never ● Only on clean exit

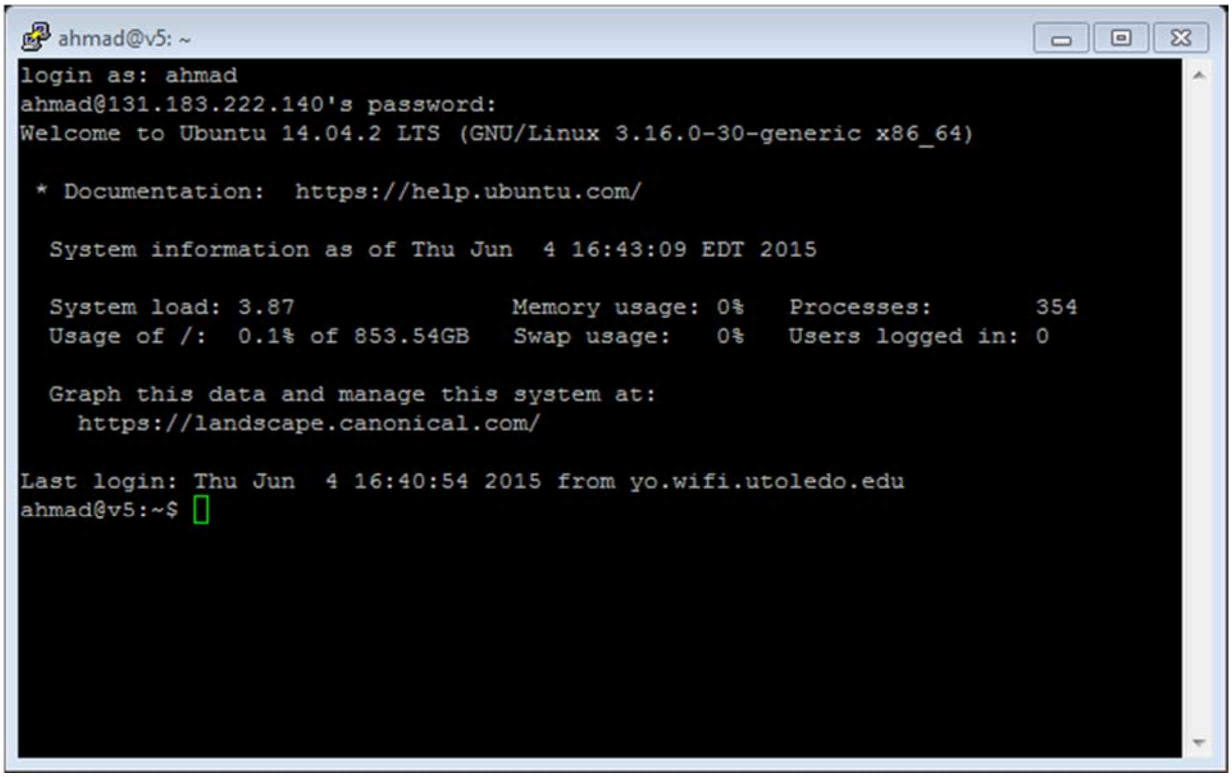
2. Step 2: Authentication

When prompted, enter the login username and password.



```
login as: 
```

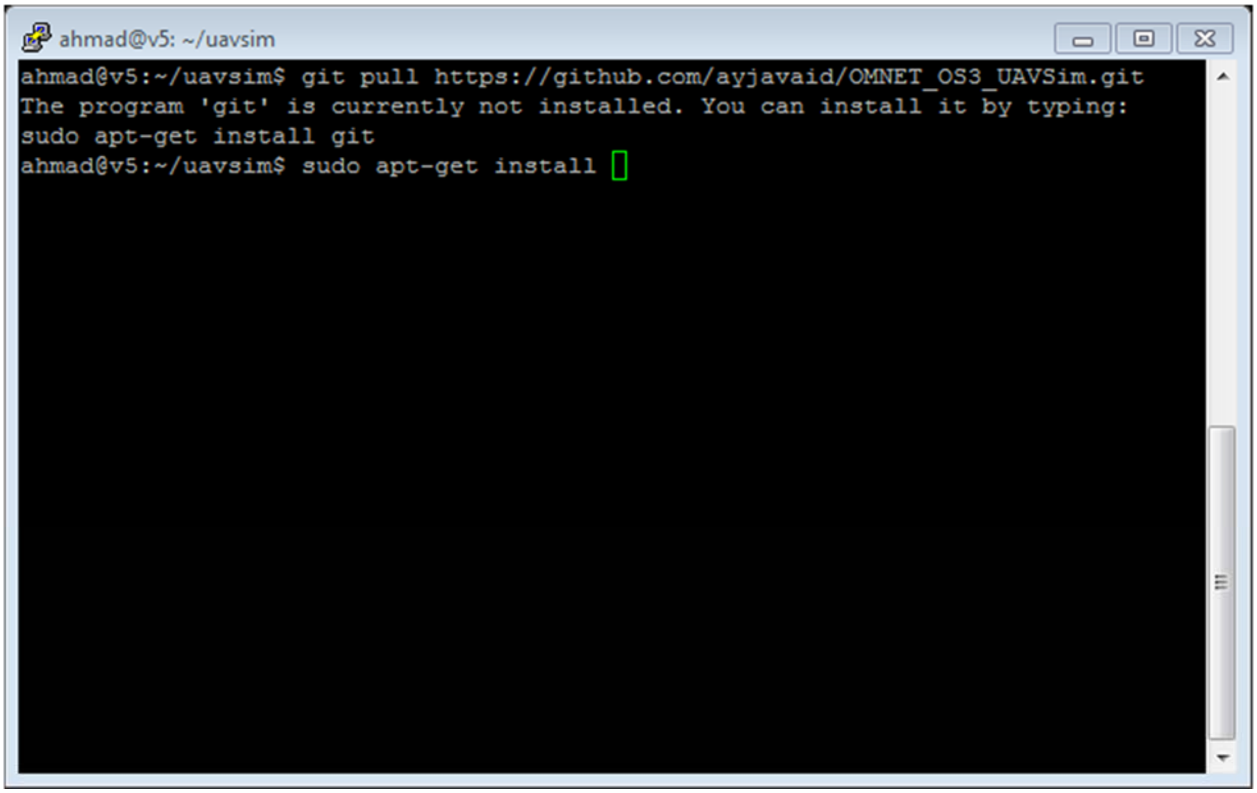
Once logged in, a screen should appear as follows:



```
login as: ahmad
ahmad@131.183.222.140's password:
Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.16.0-30-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

  System information as of Thu Jun  4 16:43:09 EDT 2015

  System load: 3.87               Memory usage: 0%   Processes:       354
  Usage of /:  0.1% of 853.54GB   Swap usage:   0%   Users logged in: 0

  Graph this data and manage this system at:
    https://landscape.canonical.com/

Last login: Thu Jun  4 16:40:54 2015 from yo.wifi.utoledo.edu
ahmad@v5:~$
```

3. Step 3. Installing git

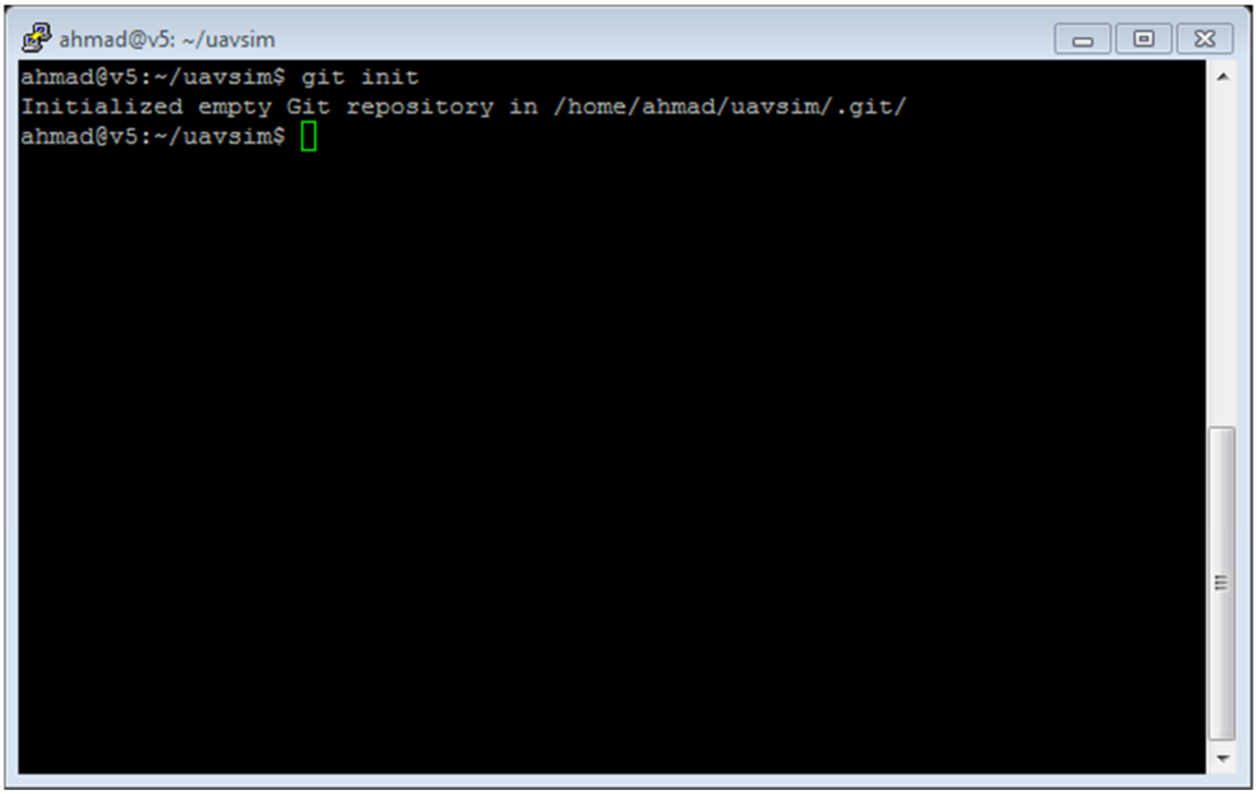
Make sure that you install git beforehand in order to clone the repository:



```
ahmad@v5:~/uavsim$ git pull https://github.com/ayjavaid/OMNET_OS3_UAVSim.git
The program 'git' is currently not installed. You can install it by typing:
sudo apt-get install git
ahmad@v5:~/uavsim$ sudo apt-get install 
```

First make a folder called uavsim and initialize a git repository in that folder.

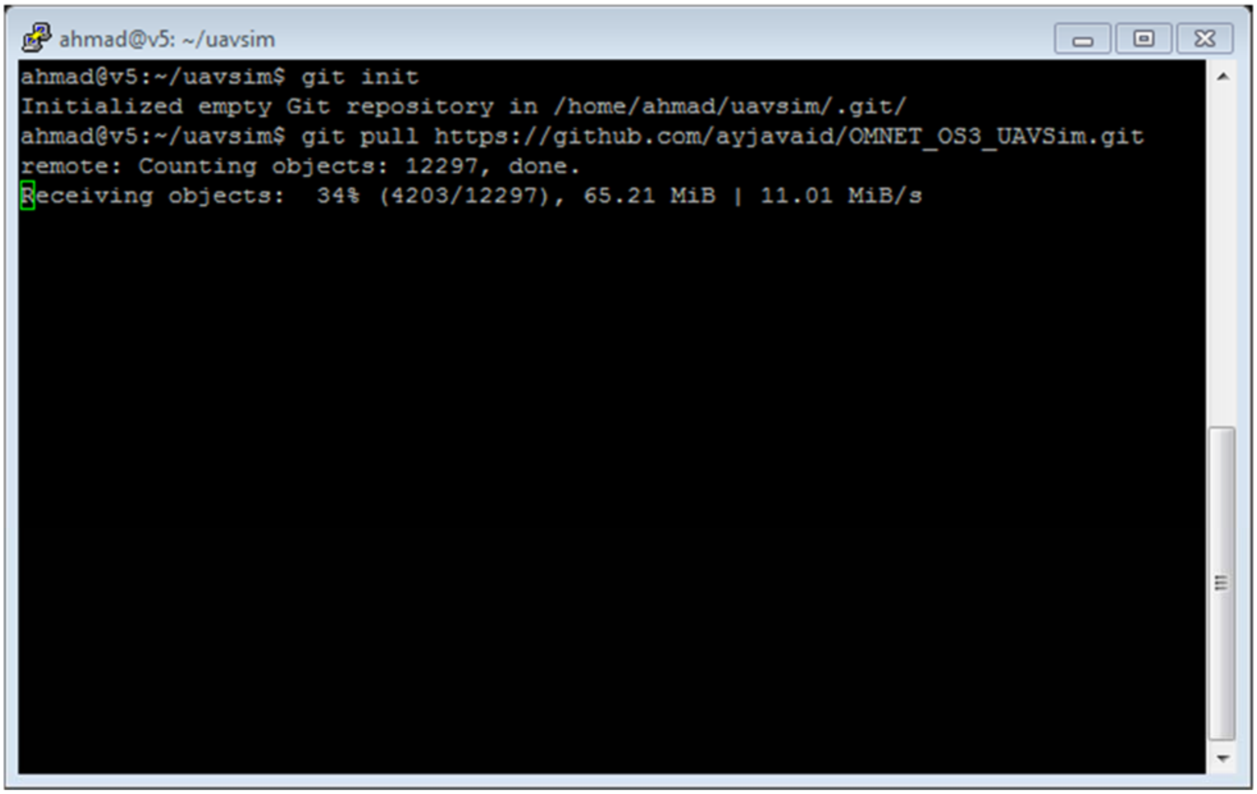
```
mkdir uavsim
git init
```



```
ahmad@v5:~/uavsim$ git init
Initialized empty Git repository in /home/ahmad/uavsim/.git/
ahmad@v5:~/uavsim$ 
```

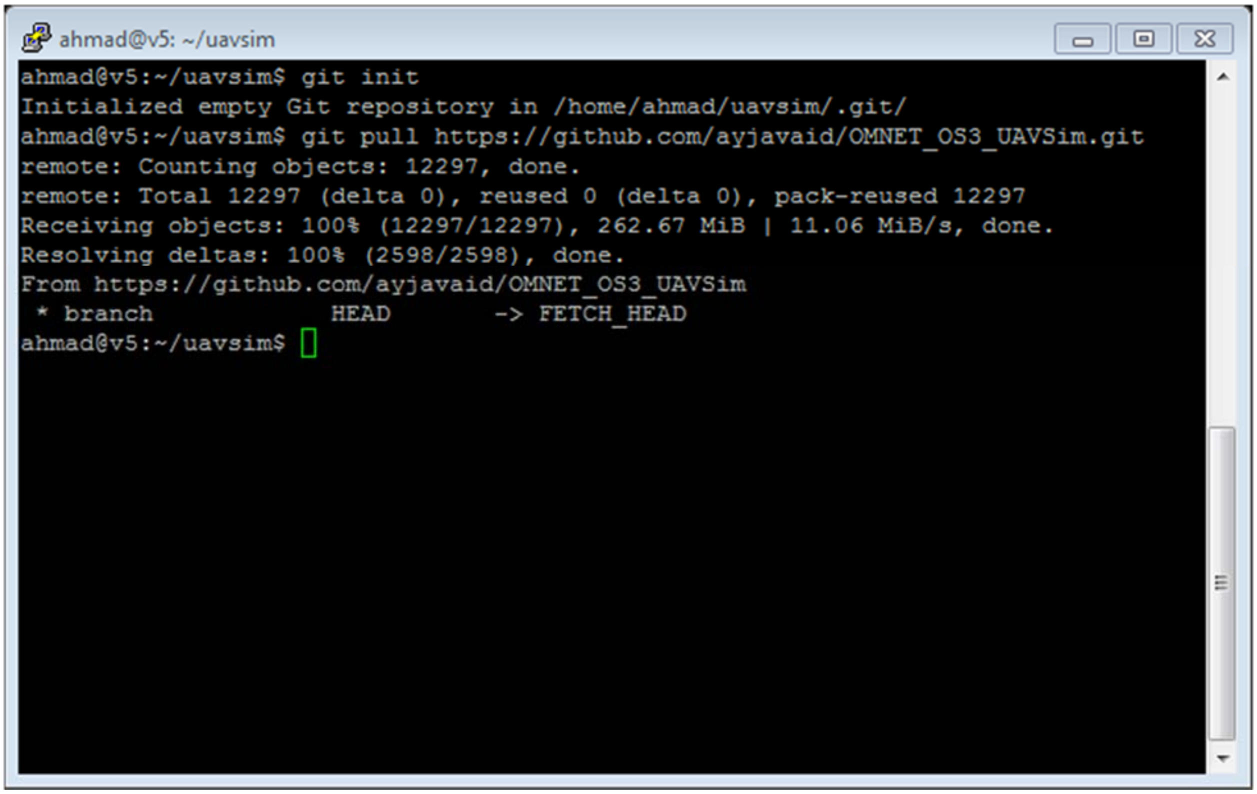
4. Step 4. Cloning the UAVSim Repository

Use the github link (https://github.com/ayjavaid/OMNET_OS3_UAVSim.git) for the open source UAVSim to "clone" the repository as shown below:



```
ahmad@v5:~/uavsim$ git init
Initialized empty Git repository in /home/ahmad/uavsim/.git/
ahmad@v5:~/uavsim$ git pull https://github.com/ayjavaid/OMNET_OS3_UAVSim.git
remote: Counting objects: 12297, done.
Receiving objects:  34% (4203/12297), 65.21 MiB | 11.01 MiB/s
```

You should get the following message once it is done:

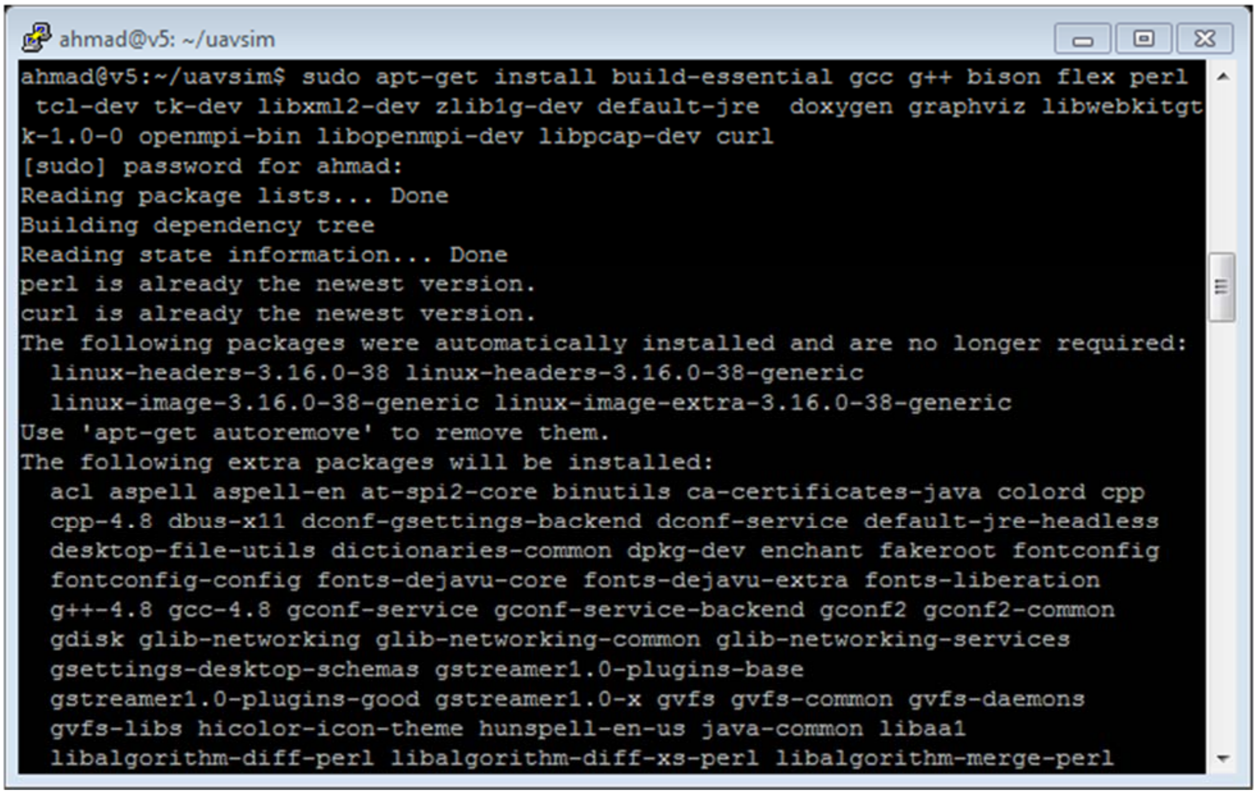


```
ahmad@v5:~/uavsim$ git init
Initialized empty Git repository in /home/ahmad/uavsim/.git/
ahmad@v5:~/uavsim$ git pull https://github.com/ayjavaid/OMNET_OS3_UAVSim.git
remote: Counting objects: 12297, done.
remote: Total 12297 (delta 0), reused 0 (delta 0), pack-reused 12297
Receiving objects: 100% (12297/12297), 262.67 MiB | 11.06 MiB/s, done.
Resolving deltas: 100% (2598/2598), done.
From https://github.com/ayjavaid/OMNET_OS3_UAVSim
 * branch            HEAD       -> FETCH_HEAD
ahmad@v5:~/uavsim$
```

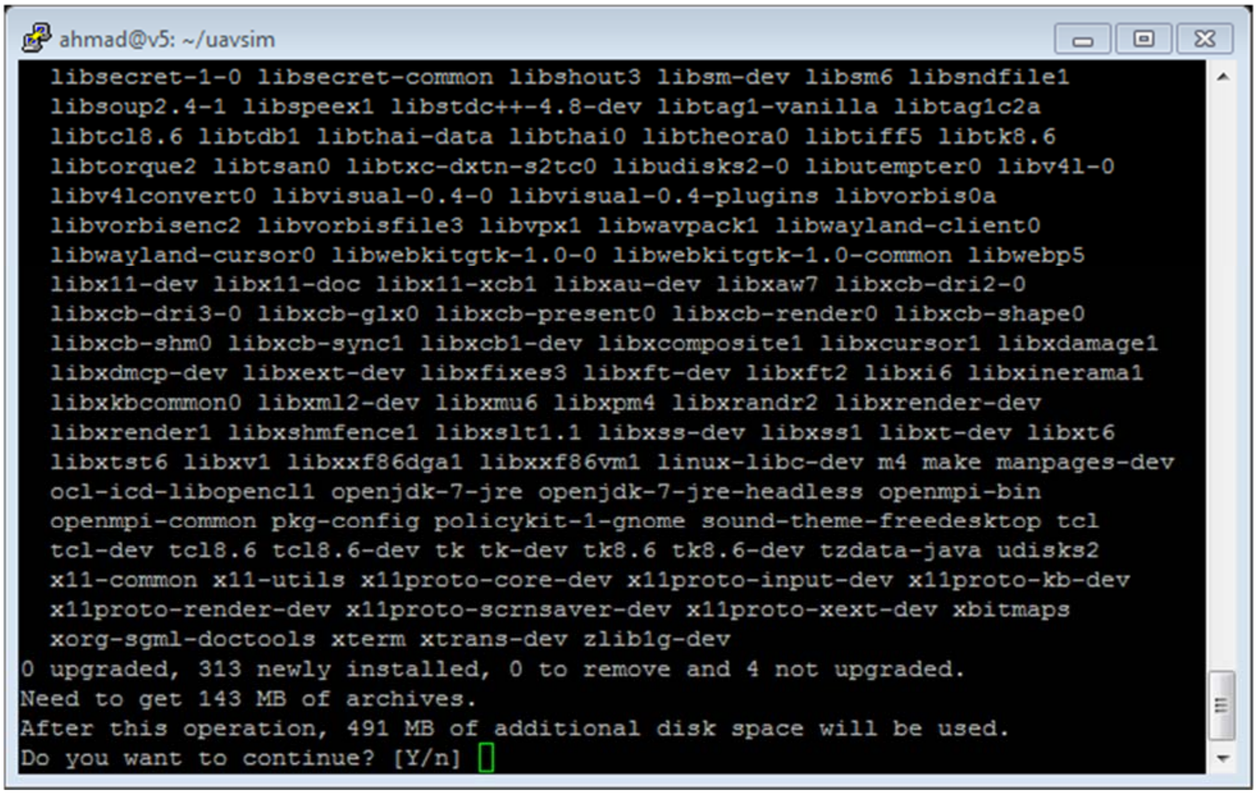
5. Step 5. UAVSim Prerequisite installations

If you already know how to install omnet++ (https://omnetpp.org/doc/omnetpp/InstallGuide.pdf) OR (http://pharos.ece.utexas.edu/wiki/index.php/How_to_Install_OMNeT%2B%2B_4.2.2_on_Ubuntu_12.04) then you know that there are several pre-requisites of this simulation software. If you already have omnet++ installed, you will not need to go through this step of installing pre-requisite modules. Use the following command:

```
$ sudo apt-get install build-essential gcc g++ bison flex perl tcl-dev
tk-dev libxml2-dev zlib1g-dev default-jre doxygen graphviz
libwebkitgtk-1.0-0 openmpi-bin libopenmpi-dev libpcap-dev libcurl-nss-
dev
```



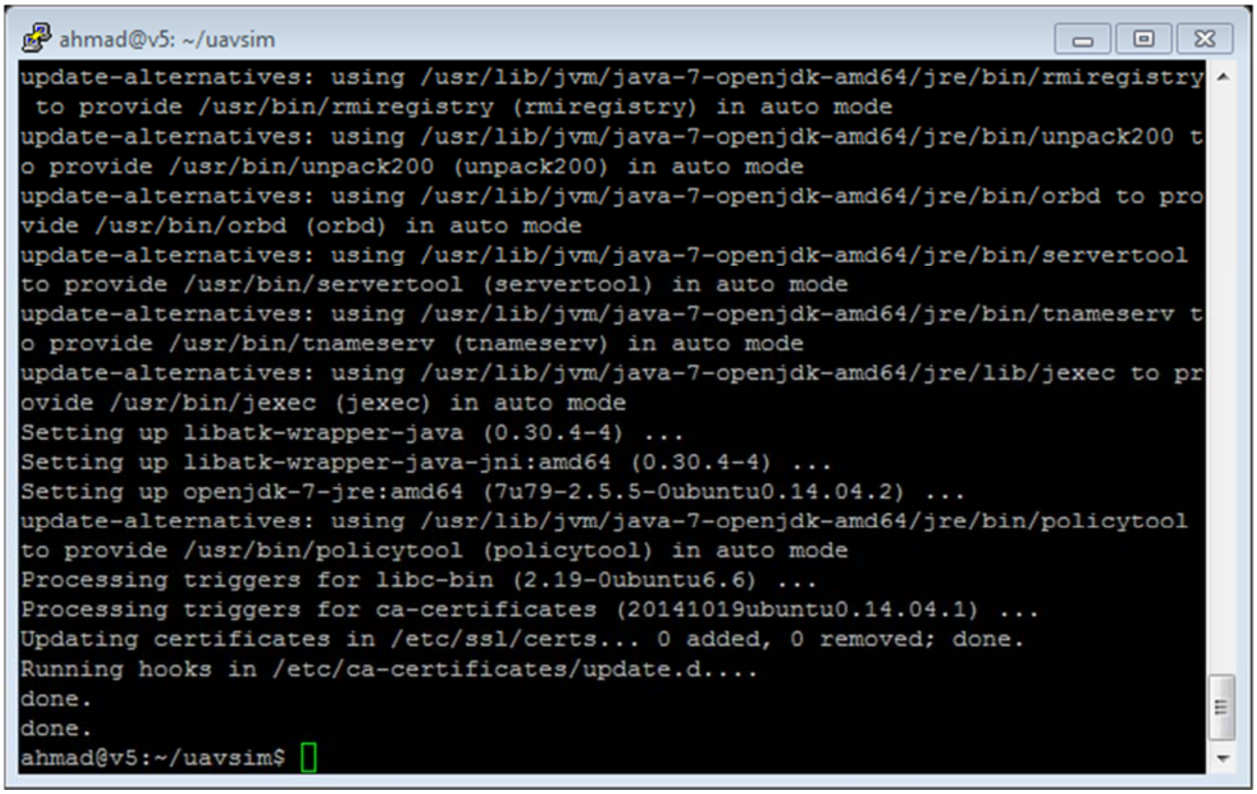
```
ahmad@v5:~/uavsim$ sudo apt-get install build-essential gcc g++ bison flex perl
 tcl-dev tk-dev libxml2-dev zlib1g-dev default-jre  doxygen graphviz libwebkitgt
k-1.0-0 openmpi-bin libopenmpi-dev libpcap-dev curl
[sudo] password for ahmad:
Reading package lists... Done
Building dependency tree
Reading state information... Done
perl is already the newest version.
curl is already the newest version.
The following packages were automatically installed and are no longer required:
  linux-headers-3.16.0-38 linux-headers-3.16.0-38-generic
  linux-image-3.16.0-38-generic linux-image-extra-3.16.0-38-generic
Use 'apt-get autoremove' to remove them.
The following extra packages will be installed:
  acl aspell aspell-en at-spi2-core binutils ca-certificates-java colord cpp
  cpp-4.8 dbus-x11 dconf-gsettings-backend dconf-service default-jre-headless
  desktop-file-utils dictionaries-common dpkg-dev enchant fakeroot fontconfig
  fontconfig-config fonts-dejavu-core fonts-dejavu-extra fonts-liberation
  g++-4.8 gcc-4.8 gconf-service gconf-service-backend gconf2 gconf2-common
  gdisk glib-networking glib-networking-common glib-networking-services
  gsettings-desktop-schemas gstreamer1.0-plugins-base
  gstreamer1.0-plugins-good gstreamer1.0-x gvfs gvfs-common gvfs-daemons
  gvfs-libs hicolor-icon-theme hunspell-en-us java-common libaa1
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl
```



```
  libsecret-1-0 libsecret-common libshout3 libsm-dev libsm6 libsndfile1
  libsoup2.4-1 libspeex1 libstdc++-4.8-dev libtag1-vanilla libtag1c2a
  libtcl8.6 libtdb1 libthai-data libthai0 libtheora0 libtiff5 libtk8.6
  libtorque2 libtsan0 libtxc-dxtn-s2tc0 libudisks2-0 libutempter0 libv4l-0
  libv4lconvert0 libvisual-0.4-0 libvisual-0.4-plugins libvorbis0a
  libvorbisenc2 libvorbisfile3 libvpx1 libwavpack1 libwayland-client0
  libwayland-cursor0 libwebkitgtk-1.0-0 libwebkitgtk-1.0-common libwebp5
  libx11-dev libx11-doc libx11-xcb1 libxau-dev libxaw7 libxcb-dri2-0
  libxcb-dri3-0 libxcb-glx0 libxcb-present0 libxcb-render0 libxcb-shape0
  libxcb-shm0 libxcb-sync1 libxcb1-dev libxcomposite1 libxcursor1 libxdamage1
  libxdmcp-dev libxext-dev libxfixes3 libxft-dev libxft2 libxi6 libxinerama1
  libxkbcommon0 libxml2-dev libxmu6 libxpm4 libxrandr2 libxrender-dev
  libxrender1 libxshmfence1 libxslt1.1 libxss-dev libxss1 libxt-dev libxt6
  libxtst6 libxv1 libxxf86dga1 libxxf86vm1 linux-libc-dev m4 make manpages-dev
  ocl-icd-libopencl1 openjdk-7-jre openjdk-7-jre-headless openmpi-bin
  openmpi-common pkg-config policykit-1-gnome sound-theme-freedesktop tcl
  tcl-dev tcl8.6 tcl8.6-dev tk tk-dev tk8.6 tk8.6-dev tzdata-java udisks2
  x11-common x11-utils x11proto-core-dev x11proto-input-dev x11proto-kb-dev
  x11proto-render-dev x11proto-scrnsaver-dev x11proto-xext-dev xbitmaps
  xorg-sgml-doctools xterm xtrans-dev zlib1g-dev
0 upgraded, 313 newly installed, 0 to remove and 4 not upgraded.
Need to get 143 MB of archives.
After this operation, 491 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

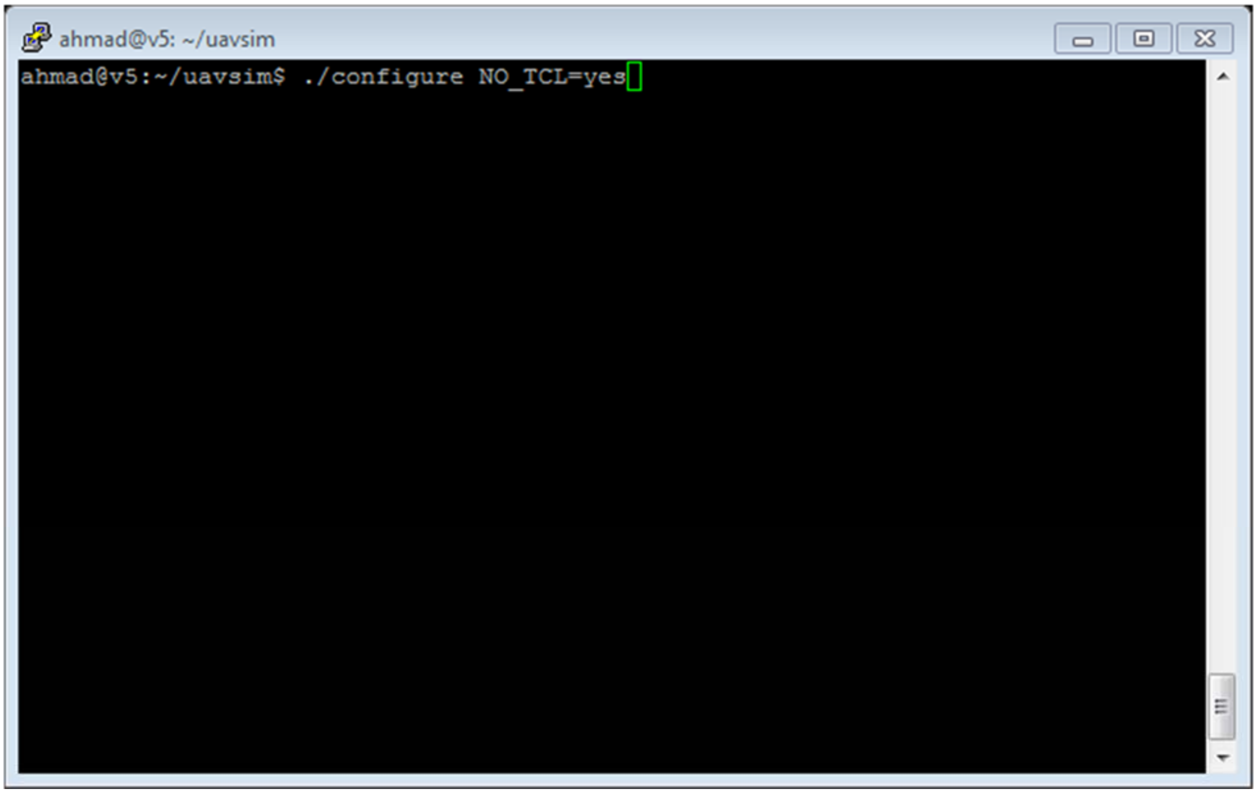
Press Y and continue the installation of these packages. It might take some depending upon your internet connection.

The following screen should appear once the installation of prerequisite packages is complete:

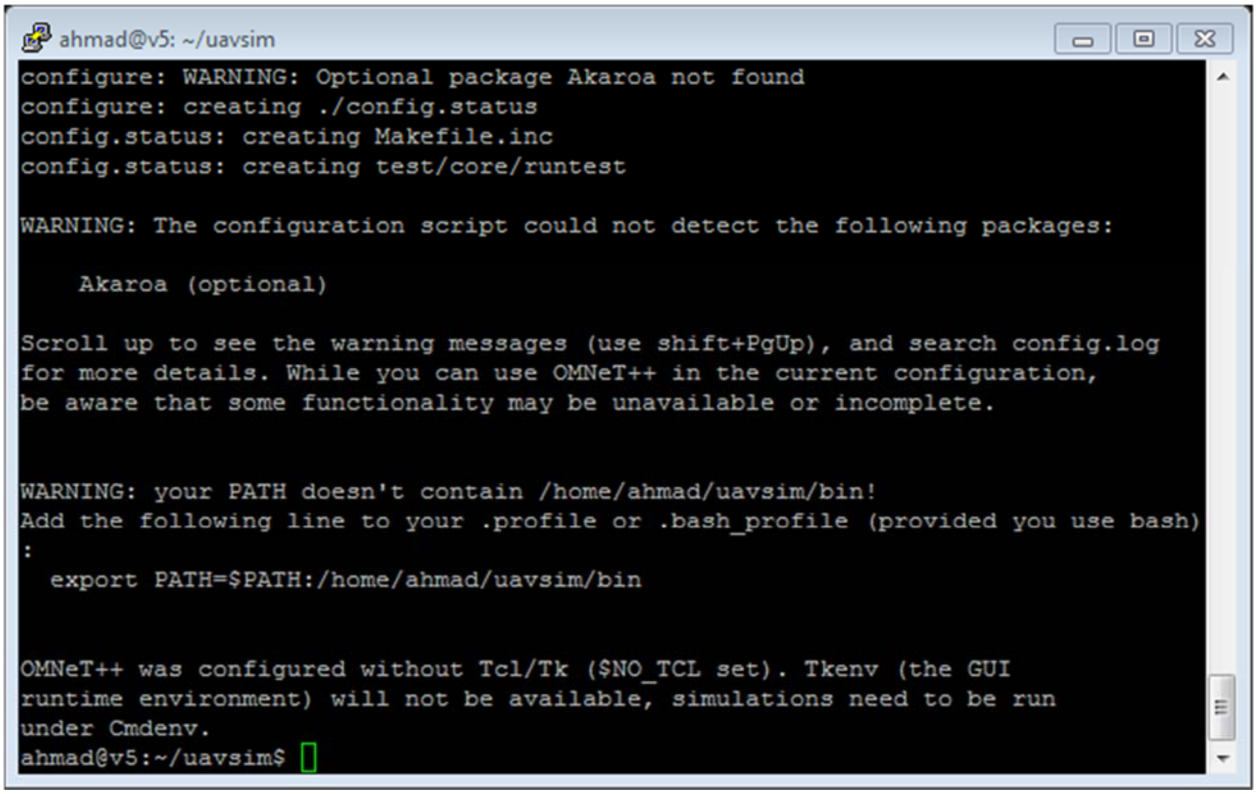


```
update-alternatives: using /usr/lib/jvm/java-7-openjdk-amd64/jre/bin/rmiregistry
 to provide /usr/bin/rmiregistry (rmiregistry) in auto mode
update-alternatives: using /usr/lib/jvm/java-7-openjdk-amd64/jre/bin/unpack200 t
o provide /usr/bin/unpack200 (unpack200) in auto mode
update-alternatives: using /usr/lib/jvm/java-7-openjdk-amd64/jre/bin/orbd to pro
vide /usr/bin/orbd (orbd) in auto mode
update-alternatives: using /usr/lib/jvm/java-7-openjdk-amd64/jre/bin/servertool
to provide /usr/bin/servertool (servertool) in auto mode
update-alternatives: using /usr/lib/jvm/java-7-openjdk-amd64/jre/bin/tnameserv t
o provide /usr/bin/tnameserv (tnameserv) in auto mode
update-alternatives: using /usr/lib/jvm/java-7-openjdk-amd64/jre/lib/jexec to pr
ovide /usr/bin/jexec (jexec) in auto mode
Setting up libatk-wrapper-java (0.30.4-4) ...
Setting up libatk-wrapper-java-jni:amd64 (0.30.4-4) ...
Setting up openjdk-7-jre:amd64 (7u79-2.5.5-0ubuntu0.14.04.2) ...
update-alternatives: using /usr/lib/jvm/java-7-openjdk-amd64/jre/bin/policytool
to provide /usr/bin/policytool (policytool) in auto mode
Processing triggers for libc-bin (2.19-0ubuntu6.6) ...
Processing triggers for ca-certificates (20141019ubuntu0.14.04.1) ...
Updating certificates in /etc/ssl/certs... 0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d....
done.
done.
ahmad@v5:~/uavsim$
```

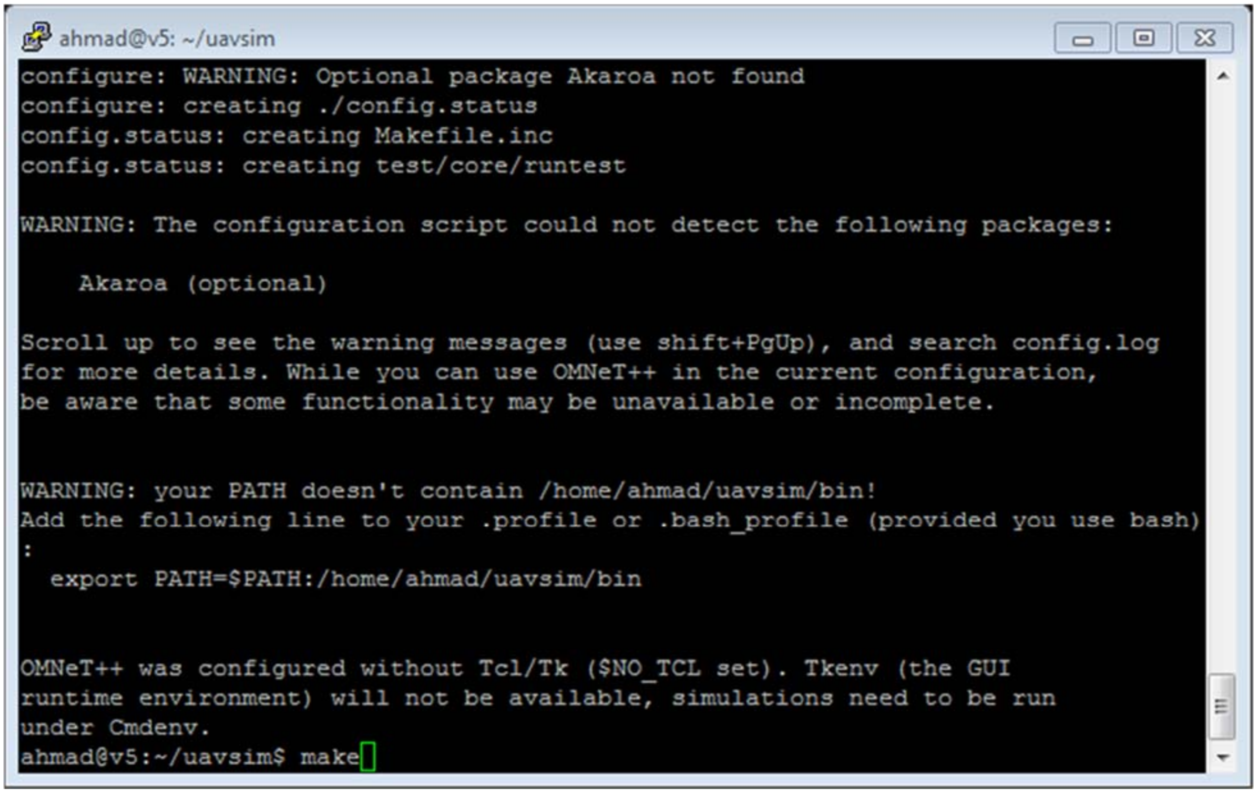
6. Step 6. UAVSim Installation



```
ahmad@v5:~/uavsim$ ./configure NO_TCL=yes
```



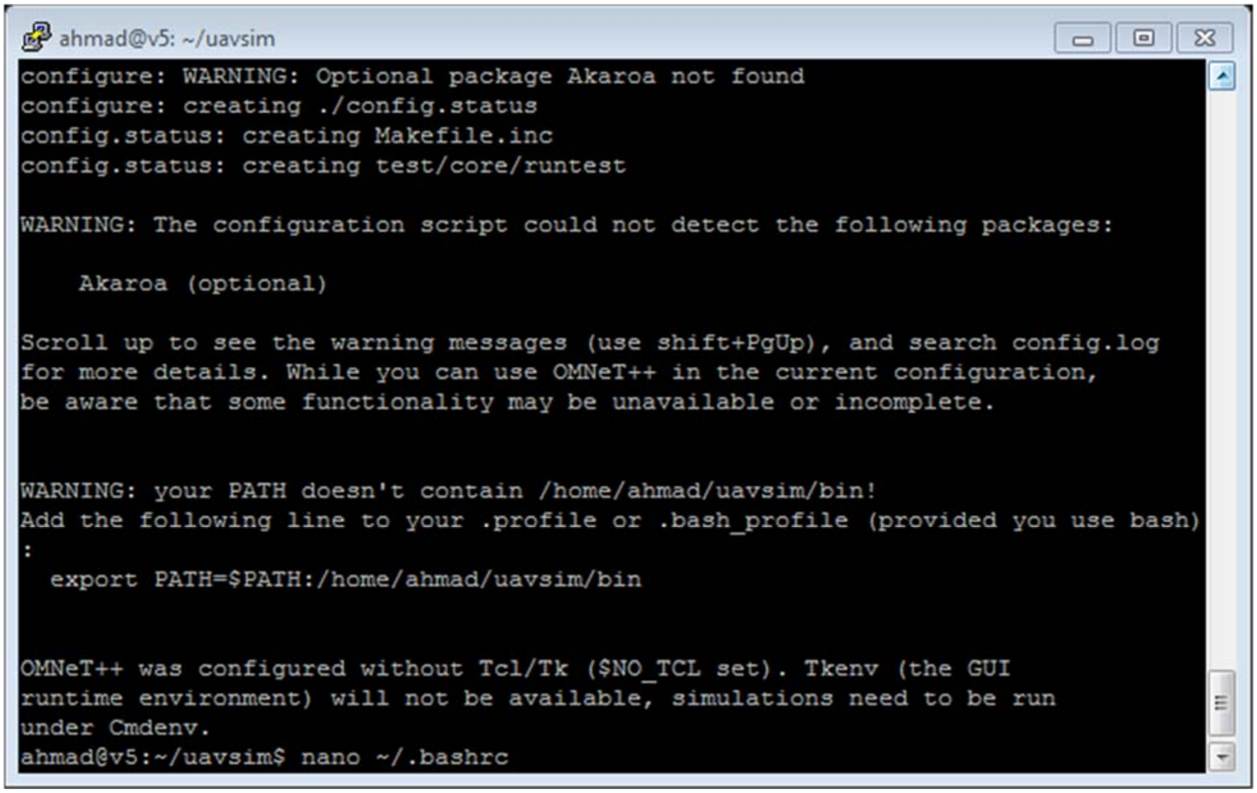
```
configure: WARNING: Optional package Akaroa not found
configure: creating ./config.status
config.status: creating Makefile.inc
config.status: creating test/core/runtest

WARNING: The configuration script could not detect the following packages:

    Akaroa (optional)

Scroll up to see the warning messages (use shift+PgUp), and search config.log
for more details. While you can use OMNeT++ in the current configuration,
be aware that some functionality may be unavailable or incomplete.


WARNING: your PATH doesn't contain /home/ahmad/uavsim/bin!
Add the following line to your .profile or .bash_profile (provided you use bash)
:
  export PATH=$PATH:/home/ahmad/uavsim/bin


OMNeT++ was configured without Tcl/Tk ($NO_TCL set). Tkenv (the GUI
runtime environment) will not be available, simulations need to be run
under Cmdenv.
ahmad@v5:~/uavsim$
```



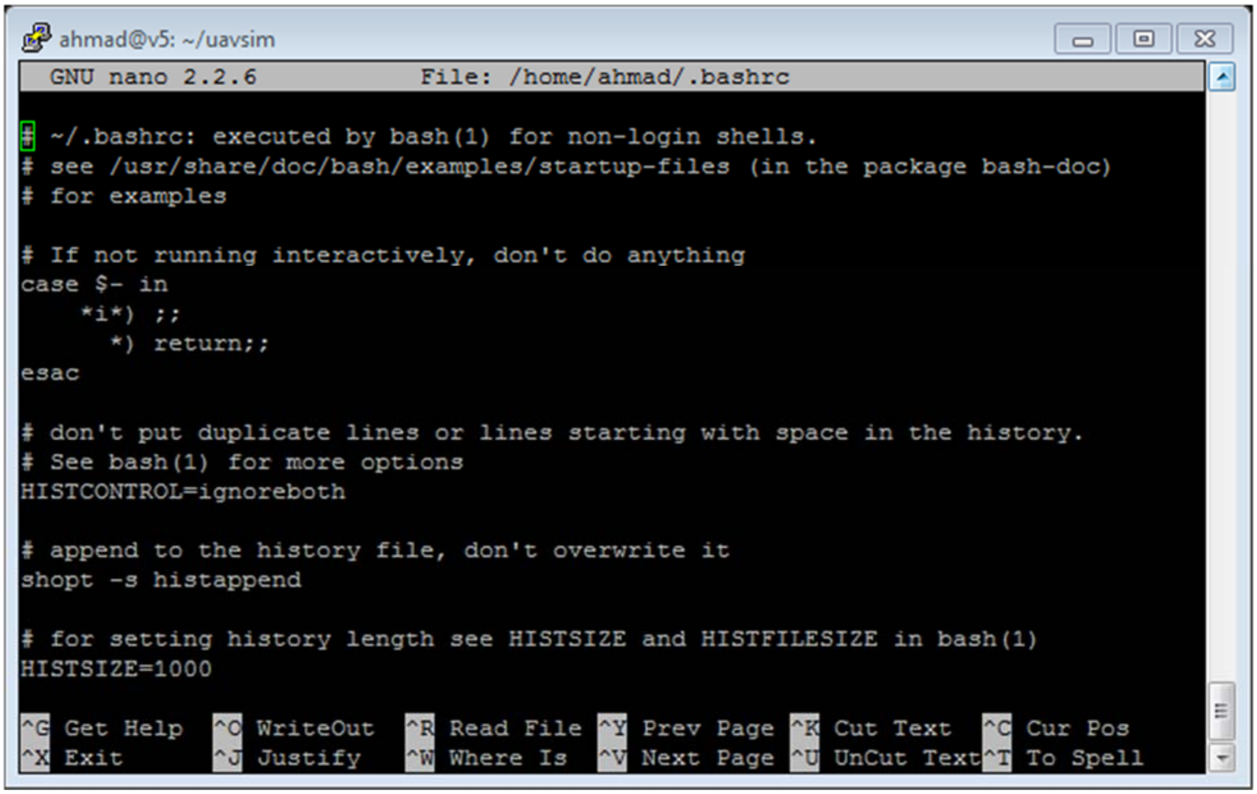
```
configure: WARNING: Optional package Akaroa not found
configure: creating ./config.status
config.status: creating Makefile.inc
config.status: creating test/core/runtest

WARNING: The configuration script could not detect the following packages:

    Akaroa (optional)

Scroll up to see the warning messages (use shift+PgUp), and search config.log
for more details. While you can use OMNeT++ in the current configuration,
be aware that some functionality may be unavailable or incomplete.


WARNING: your PATH doesn't contain /home/ahmad/uavsim/bin!
Add the following line to your .profile or .bash_profile (provided you use bash)
:
  export PATH=$PATH:/home/ahmad/uavsim/bin


OMNeT++ was configured without Tcl/Tk ($NO_TCL set). Tkenv (the GUI
runtime environment) will not be available, simulations need to be run
under Cmdenv.
ahmad@v5:~/uavsim$ make
```

As you see, we need to add the current uavsim to the path otherwise we would need to run omnet++ from this path every time using the console. If you don’t have any other OMNeT++ installation on your system, you can continue otherwise skip to next sub-step.

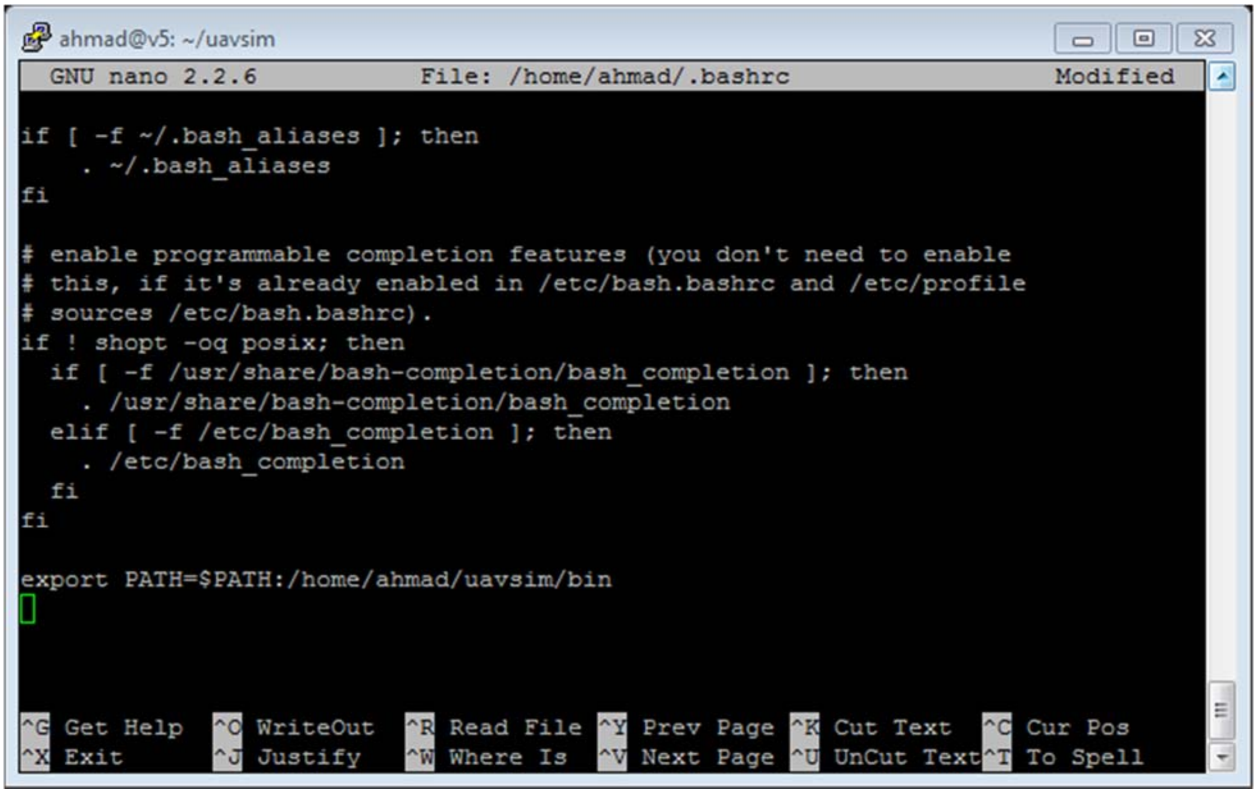
Adding uavsim/bin to path:



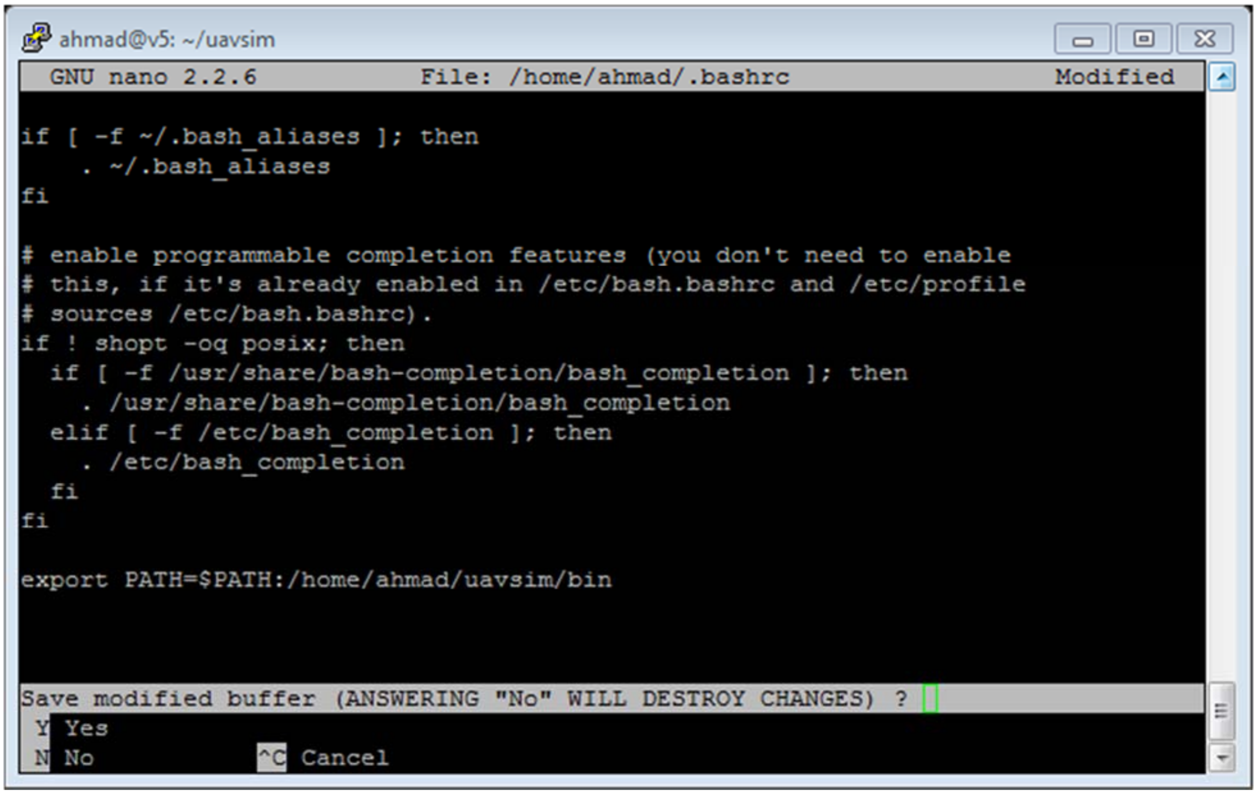
```
configure: WARNING: Optional package Akaroa not found
configure: creating ./config.status
config.status: creating Makefile.inc
config.status: creating test/core/runtest

WARNING: The configuration script could not detect the following packages:

    Akaroa (optional)

Scroll up to see the warning messages (use shift+PgUp), and search config.log
for more details. While you can use OMNeT++ in the current configuration,
be aware that some functionality may be unavailable or incomplete.


WARNING: your PATH doesn't contain /home/ahmad/uavsim/bin!
Add the following line to your .profile or .bash_profile (provided you use bash)
:
  export PATH=$PATH:/home/ahmad/uavsim/bin


OMNeT++ was configured without Tcl/Tk ($NO_TCL set). Tkenv (the GUI
runtime environment) will not be available, simulations need to be run
under Cmdenv.
ahmad@v5:~/uavsim$ nano ~/.bashrc
```



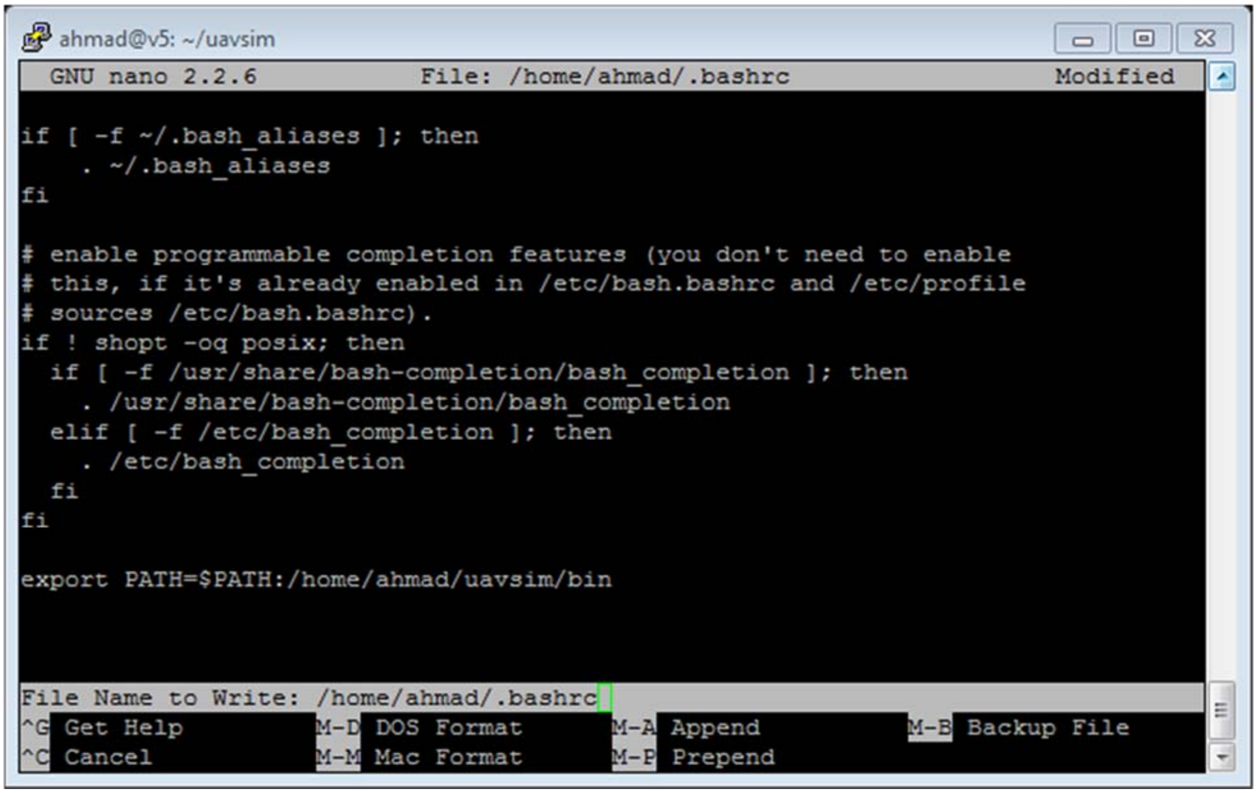
```
  GNU nano 2.2.6             File: /home/ahmad/.bashrc

# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
      *) return;;
esac

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
HISTSIZE=1000

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```



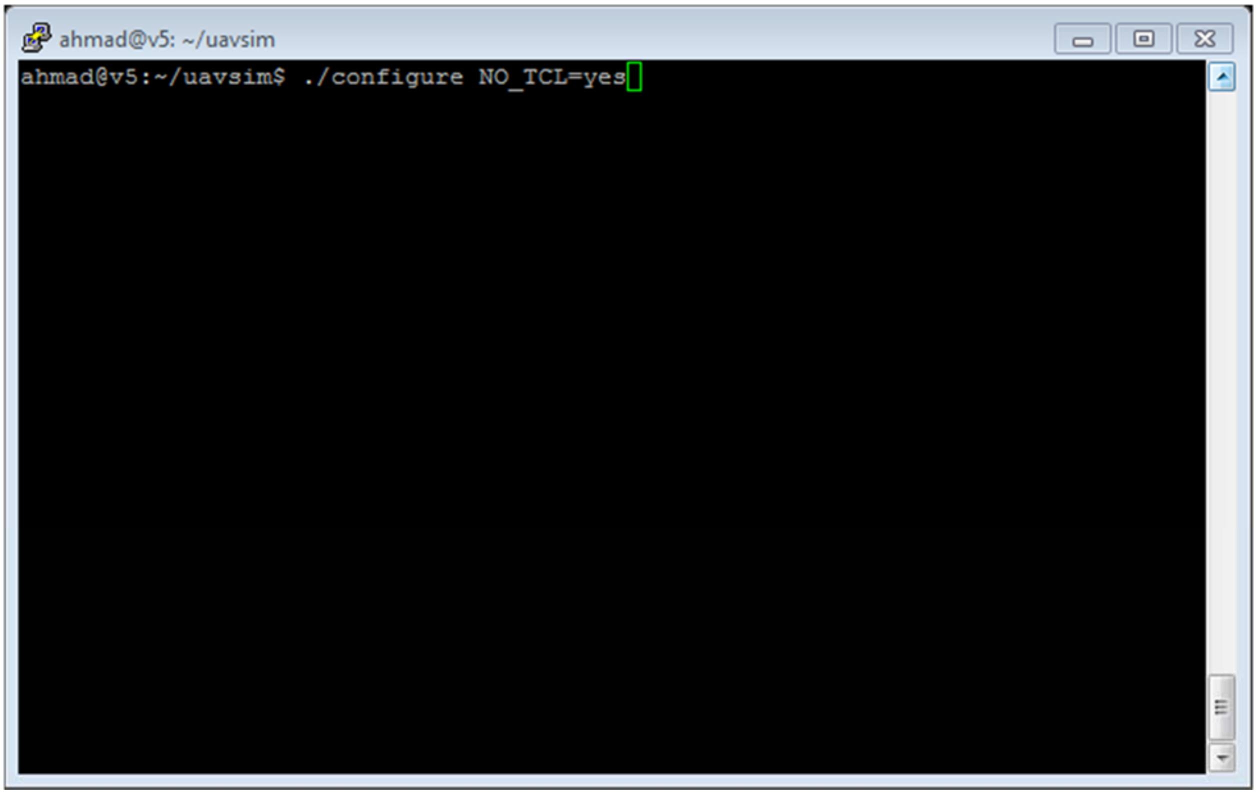
```
  GNU nano 2.2.6             File: /home/ahmad/.bashrc                Modified

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

export PATH=$PATH:/home/ahmad/uavsim/bin


^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```



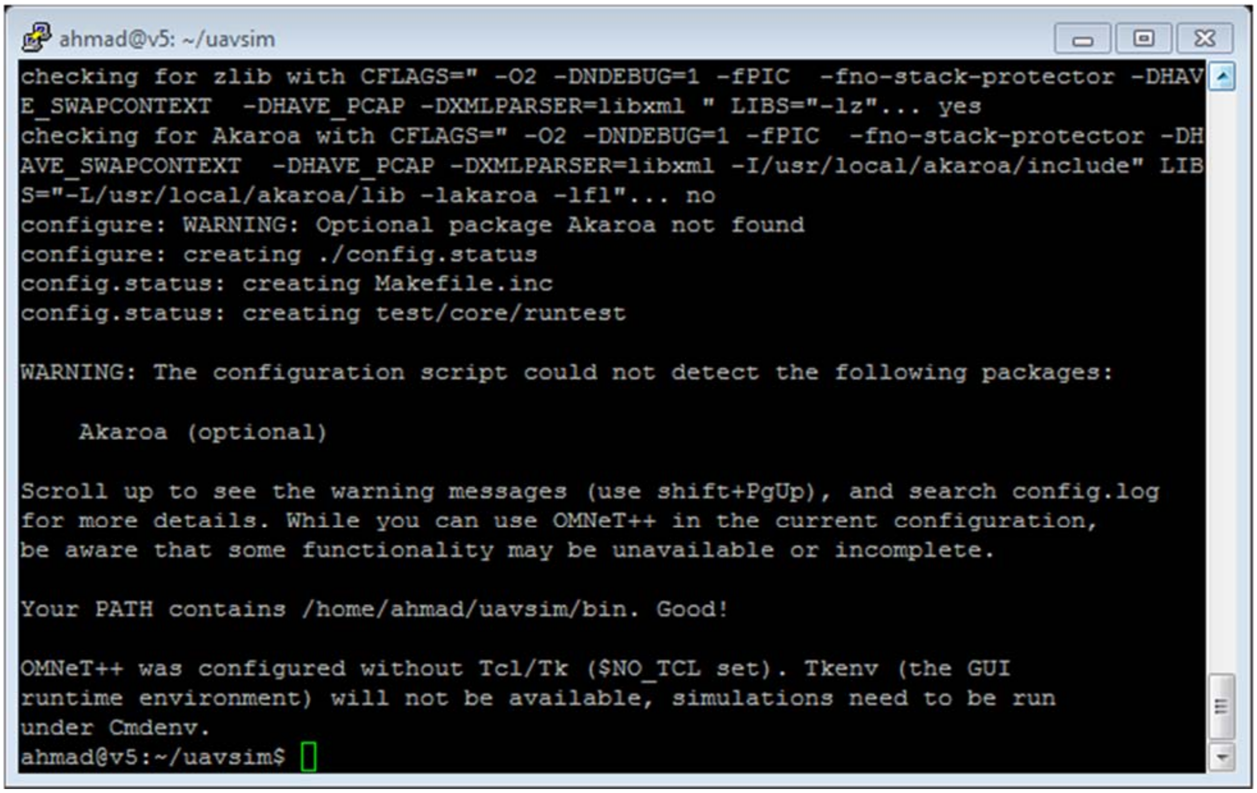
```
  GNU nano 2.2.6           File: /home/ahmad/.bashrc                  Modified

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

export PATH=$PATH:/home/ahmad/uavsim/bin



Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?
 Y Yes
 N No           ^C Cancel
```



```
  GNU nano 2.2.6           File: /home/ahmad/.bashrc                  Modified

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

export PATH=$PATH:/home/ahmad/uavsim/bin



File Name to Write: /home/ahmad/.bashrc
^G Get Help          M-D DOS Format       M-A Append           M-B Backup File
^C Cancel            M-M Mac Format       M-P Prepend
```



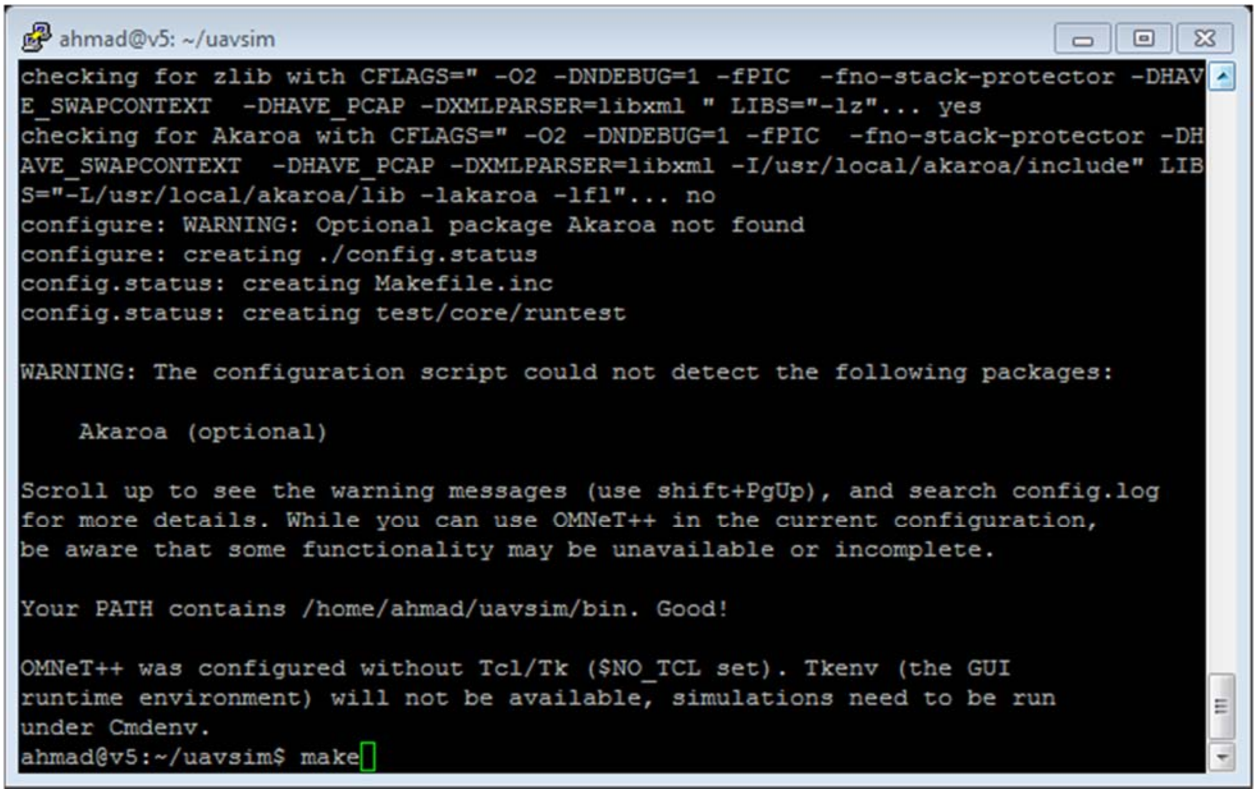
```
ahmad@v5:~/uavsim$ ./configure NO_TCL=yes
```



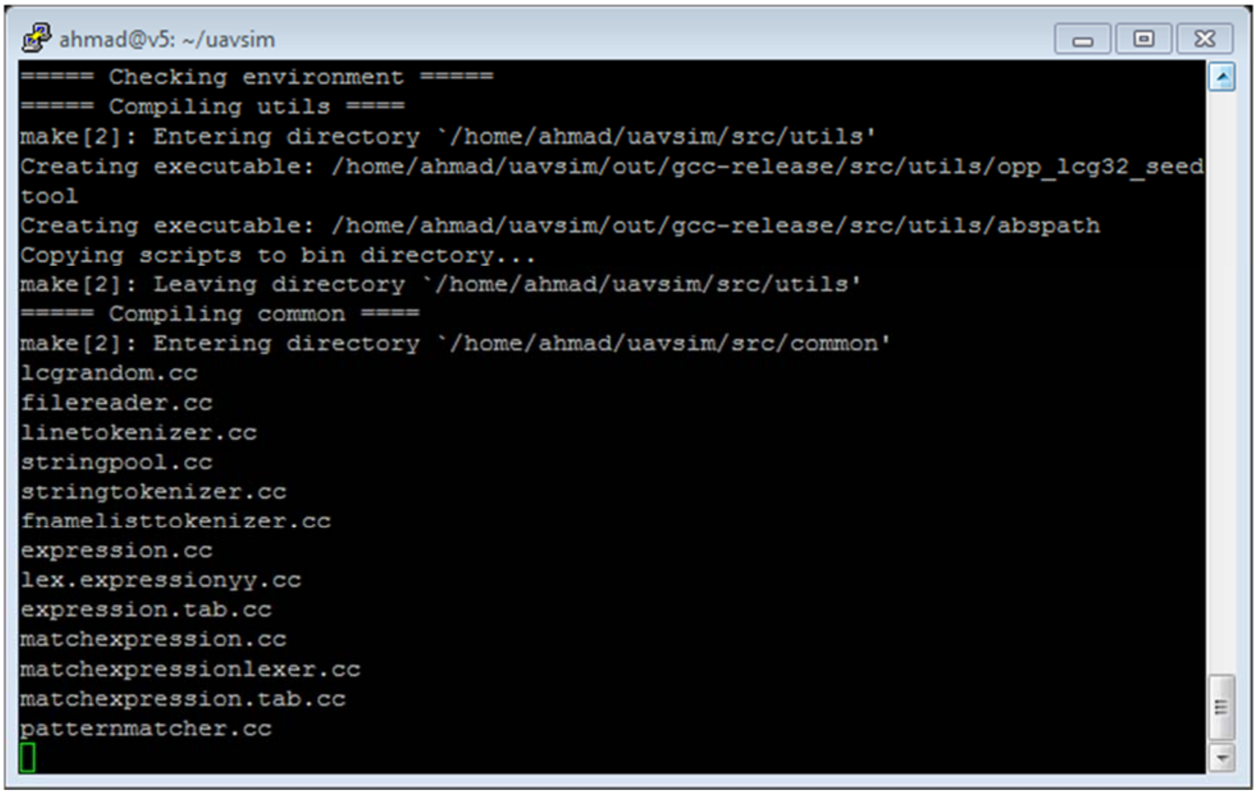
```
checking for zlib with CFLAGS=" -O2 -DNDEBUG=1 -fPIC  -fno-stack-protector -DHAV
E_SWAPCONTEXT  -DHAVE_PCAP -DXMLPARSER=libxml " LIBS="-lz"... yes
checking for Akaroa with CFLAGS=" -O2 -DNDEBUG=1 -fPIC  -fno-stack-protector -DH
AVE_SWAPCONTEXT  -DHAVE_PCAP -DXMLPARSER=libxml -I/usr/local/akaroa/include" LIB
S="-L/usr/local/akaroa/lib -lakaroa -lfl"... no
configure: WARNING: Optional package Akaroa not found
configure: creating ./config.status
config.status: creating Makefile.inc
config.status: creating test/core/runtest

WARNING: The configuration script could not detect the following packages:

    Akaroa (optional)

Scroll up to see the warning messages (use shift+PgUp), and search config.log
for more details. While you can use OMNeT++ in the current configuration,
be aware that some functionality may be unavailable or incomplete.

Your PATH contains /home/ahmad/uavsim/bin. Good!

OMNeT++ was configured without Tcl/Tk ($NO_TCL set). Tkenv (the GUI
runtime environment) will not be available, simulations need to be run
under Cmdenv.
ahmad@v5:~/uavsim$
```

If you wish, you can install the Akaroa package but its optional and would not affect the operation of UAVSim.

Final install:

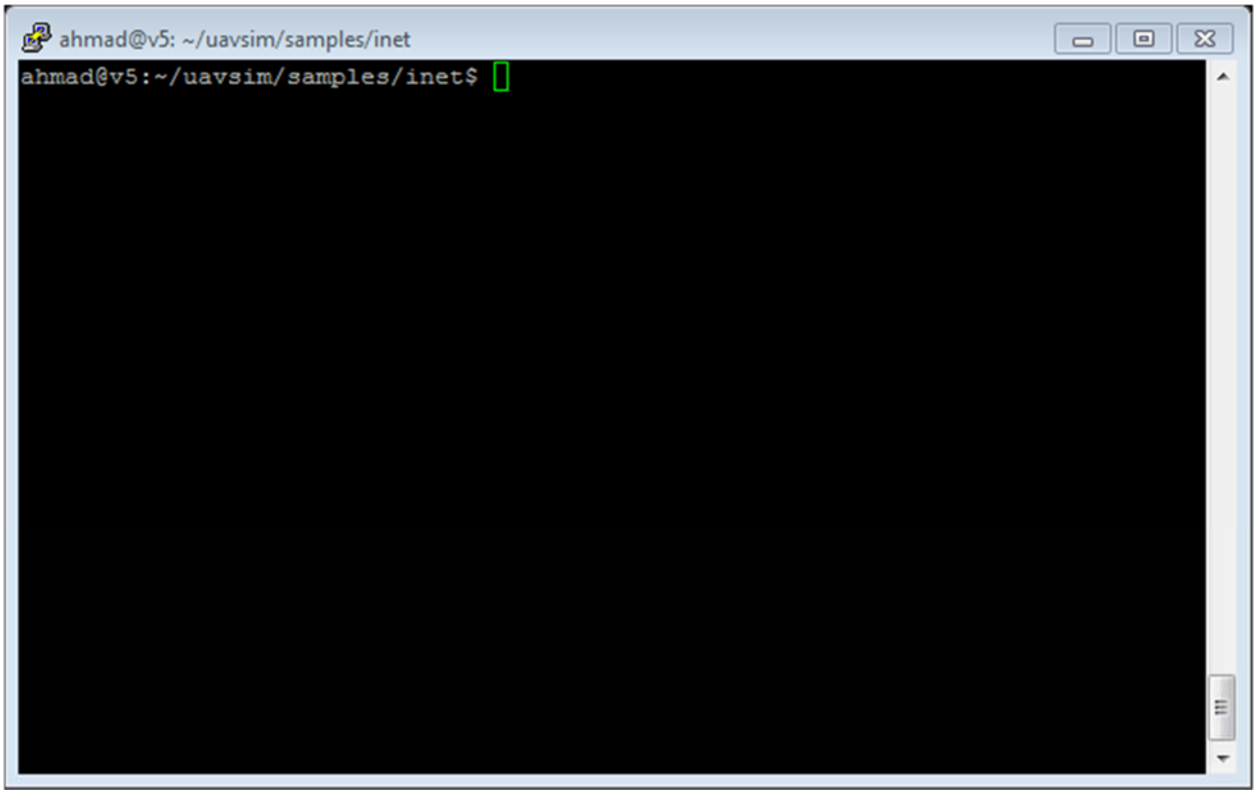


```
checking for zlib with CFLAGS=" -O2 -DNDEBUG=1 -fPIC  -fno-stack-protector -DHAV
E_SWAPCONTEXT  -DHAVE_PCAP -DXMLPARSER=libxml " LIBS="-lz"... yes
checking for Akaroa with CFLAGS=" -O2 -DNDEBUG=1 -fPIC  -fno-stack-protector -DH
AVE_SWAPCONTEXT  -DHAVE_PCAP -DXMLPARSER=libxml -I/usr/local/akaroa/include" LIB
S="-L/usr/local/akaroa/lib -lakaroa -lfl"... no
configure: WARNING: Optional package Akaroa not found
configure: creating ./config.status
config.status: creating Makefile.inc
config.status: creating test/core/runtest

WARNING: The configuration script could not detect the following packages:

    Akaroa (optional)

Scroll up to see the warning messages (use shift+PgUp), and search config.log
for more details. While you can use OMNeT++ in the current configuration,
be aware that some functionality may be unavailable or incomplete.

Your PATH contains /home/ahmad/uavsim/bin. Good!

OMNeT++ was configured without Tcl/Tk ($NO_TCL set). Tkenv (the GUI
runtime environment) will not be available, simulations need to be run
under Cmdenv.
ahmad@v5:~/uavsim$ make
```

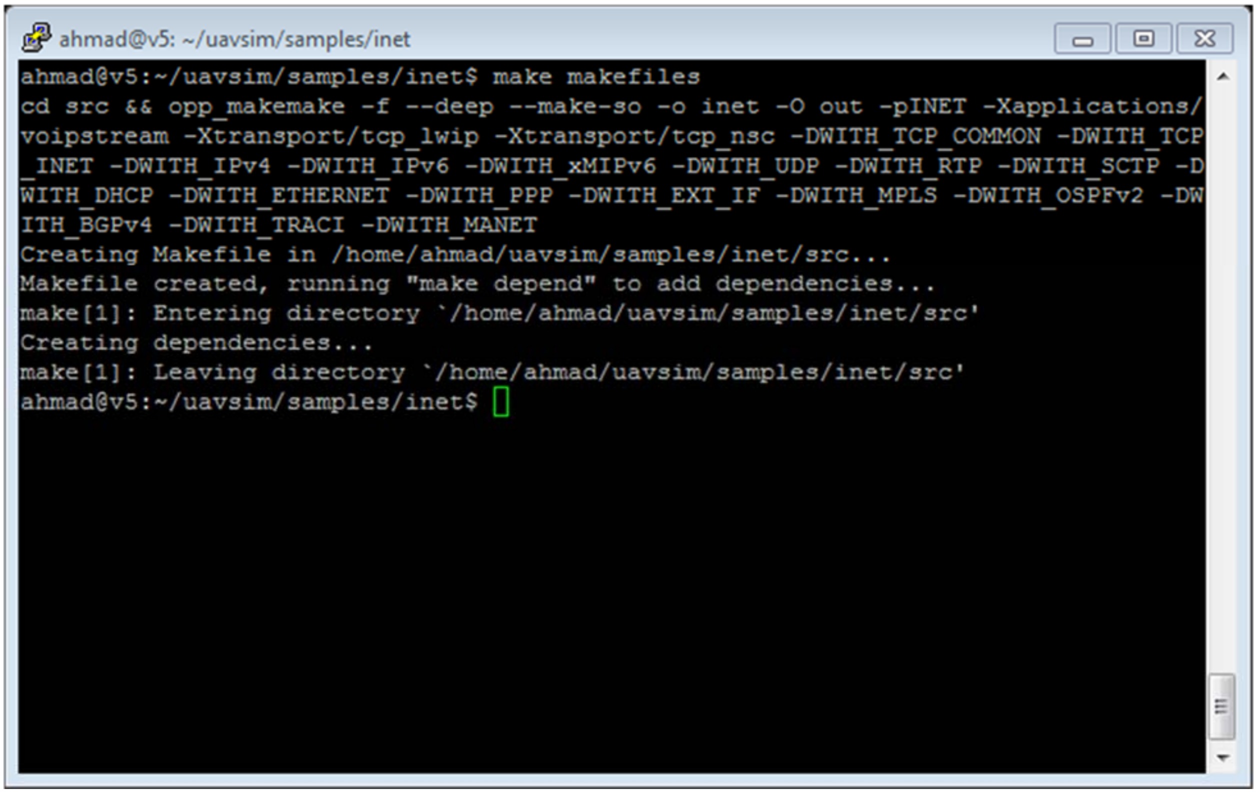


```
===== Checking environment =====
===== Compiling utils ====
make[2]: Entering directory `/home/ahmad/uavsim/src/utils'
Creating executable: /home/ahmad/uavsim/out/gcc-release/src/utils/opp_lcg32_seed
tool
Creating executable: /home/ahmad/uavsim/out/gcc-release/src/utils/abspath
Copying scripts to bin directory...
make[2]: Leaving directory `/home/ahmad/uavsim/src/utils'
===== Compiling common ====
make[2]: Entering directory `/home/ahmad/uavsim/src/common'
lcgrandom.cc
filereader.cc
linetokenizer.cc
stringpool.cc
stringtokenizer.cc
fnamelisttokenizer.cc
expression.cc
lex.expressionyy.cc
expression.tab.cc
matchexpression.cc
matchexpressionlexer.cc
matchexpression.tab.cc
patternmatcher.cc
```

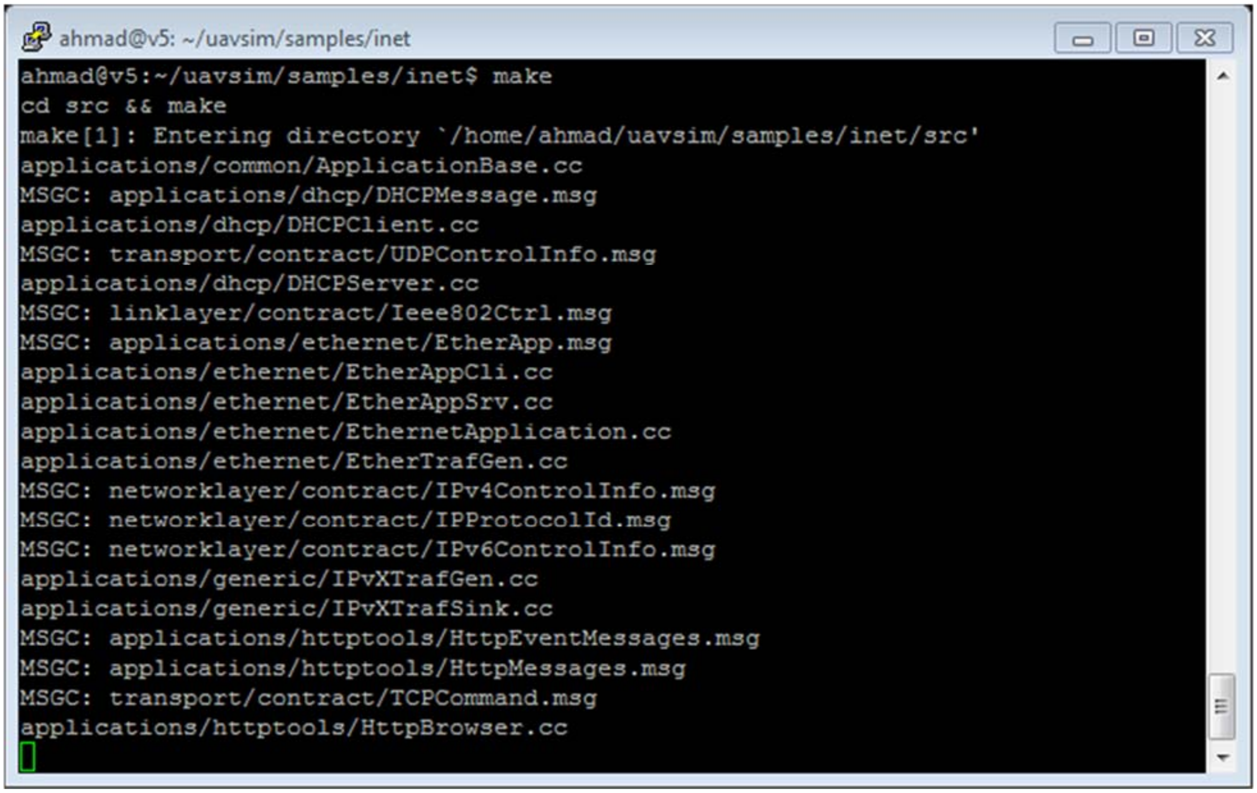
Once compiled, you might get a few errors because we have removed the basic examples included in OMNeT++. Once this is done, the inet and CNI_OS3 have to be separately compiled. For this, please go to the directory as shown below:



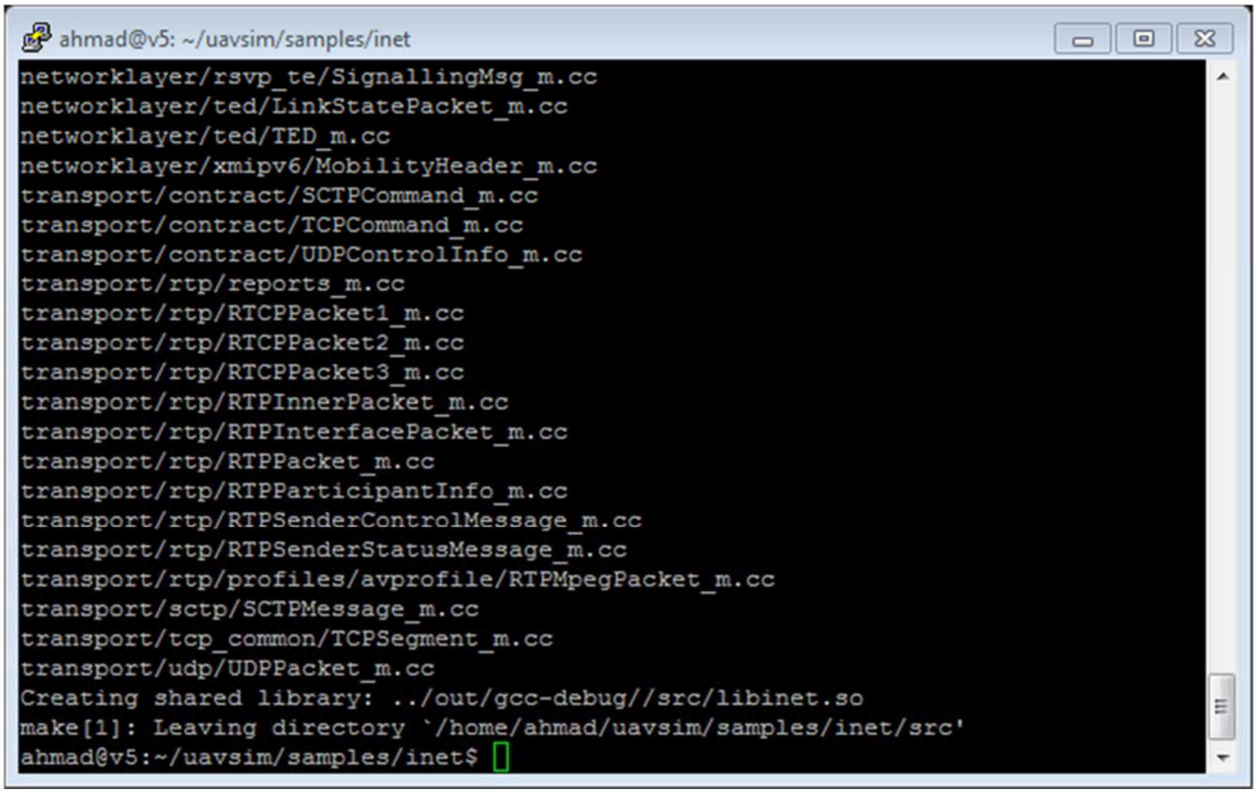
```
ahmad@v5:~/uavsim/samples/inet$
```



```
ahmad@v5:~/uavsim/samples/inet$ make makefiles
cd src && opp_makemake -f --deep --make-so -o inet -O out -pINET -Xapplications/
voipstream -Xtransport/tcp_lwip -Xtransport/tcp_nsc -DWITH_TCP_COMMON -DWITH_TCP
_INET -DWITH_IPv4 -DWITH_IPv6 -DWITH_xMIPv6 -DWITH_UDP -DWITH_RTP -DWITH_SCTP -D
WITH_DHCP -DWITH_ETHERNET -DWITH_PPP -DWITH_EXT_IF -DWITH_MPLS -DWITH_OSPFv2 -DW
ITH_BGPv4 -DWITH_TRACI -DWITH_MANET
Creating Makefile in /home/ahmad/uavsim/samples/inet/src...
Makefile created, running "make depend" to add dependencies...
make[1]: Entering directory `/home/ahmad/uavsim/samples/inet/src'
Creating dependencies...
make[1]: Leaving directory `/home/ahmad/uavsim/samples/inet/src'
ahmad@v5:~/uavsim/samples/inet$
```

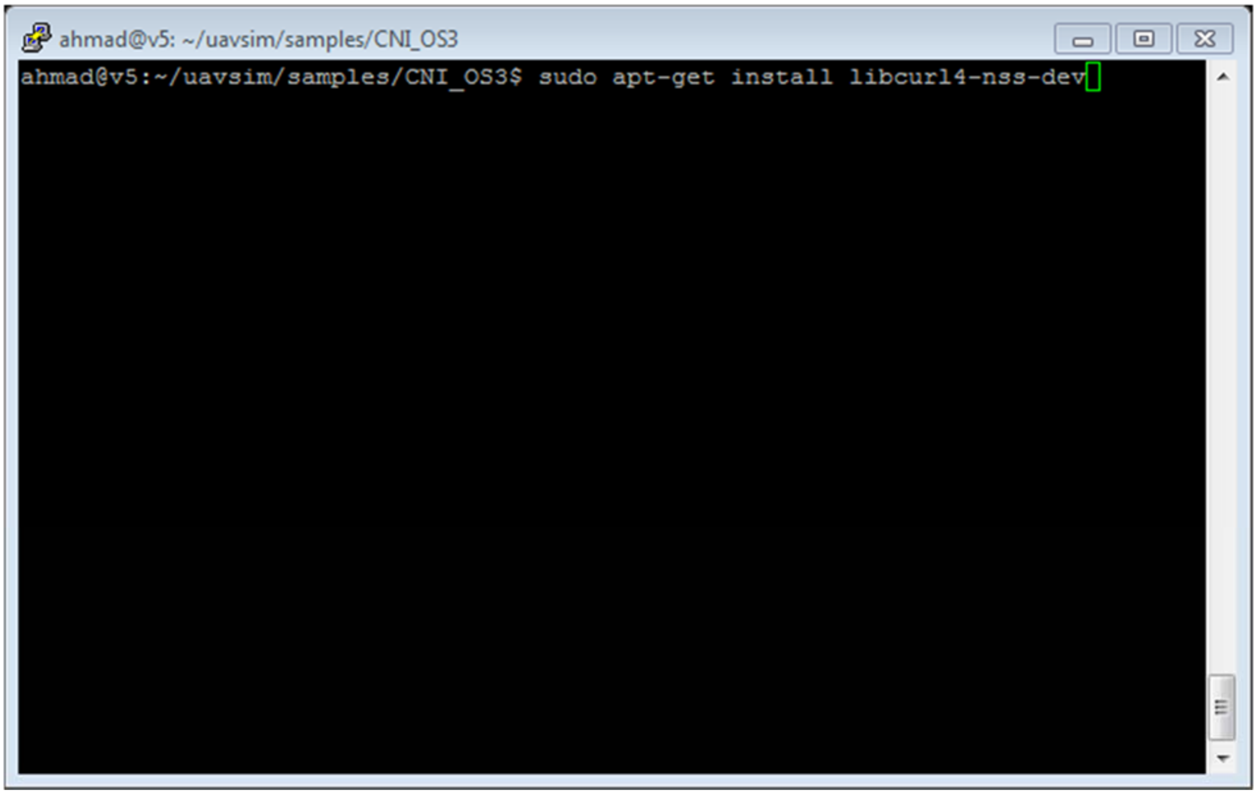


```
ahmad@v5:~/uavsim/samples/inet$ make
cd src && make
make[1]: Entering directory `/home/ahmad/uavsim/samples/inet/src'
applications/common/ApplicationBase.cc
MSGC: applications/dhcp/DHCPMessage.msg
applications/dhcp/DHCPClient.cc
MSGC: transport/contract/UDPControlInfo.msg
applications/dhcp/DHCPServer.cc
MSGC: linklayer/contract/Ieee802Ctrl.msg
MSGC: applications/ethernet/EtherApp.msg
applications/ethernet/EtherAppCli.cc
applications/ethernet/EtherAppSrv.cc
applications/ethernet/EthernetApplication.cc
applications/ethernet/EtherTrafGen.cc
MSGC: networklayer/contract/IPv4ControlInfo.msg
MSGC: networklayer/contract/IPProtocolId.msg
MSGC: networklayer/contract/IPv6ControlInfo.msg
applications/generic/IPvXTrafGen.cc
applications/generic/IPvXTrafSink.cc
MSGC: applications/httptools/HttpEventMessages.msg
MSGC: applications/httptools/HttpMessages.msg
MSGC: transport/contract/TCPCommand.msg
applications/httptools/HttpBrowser.cc
```

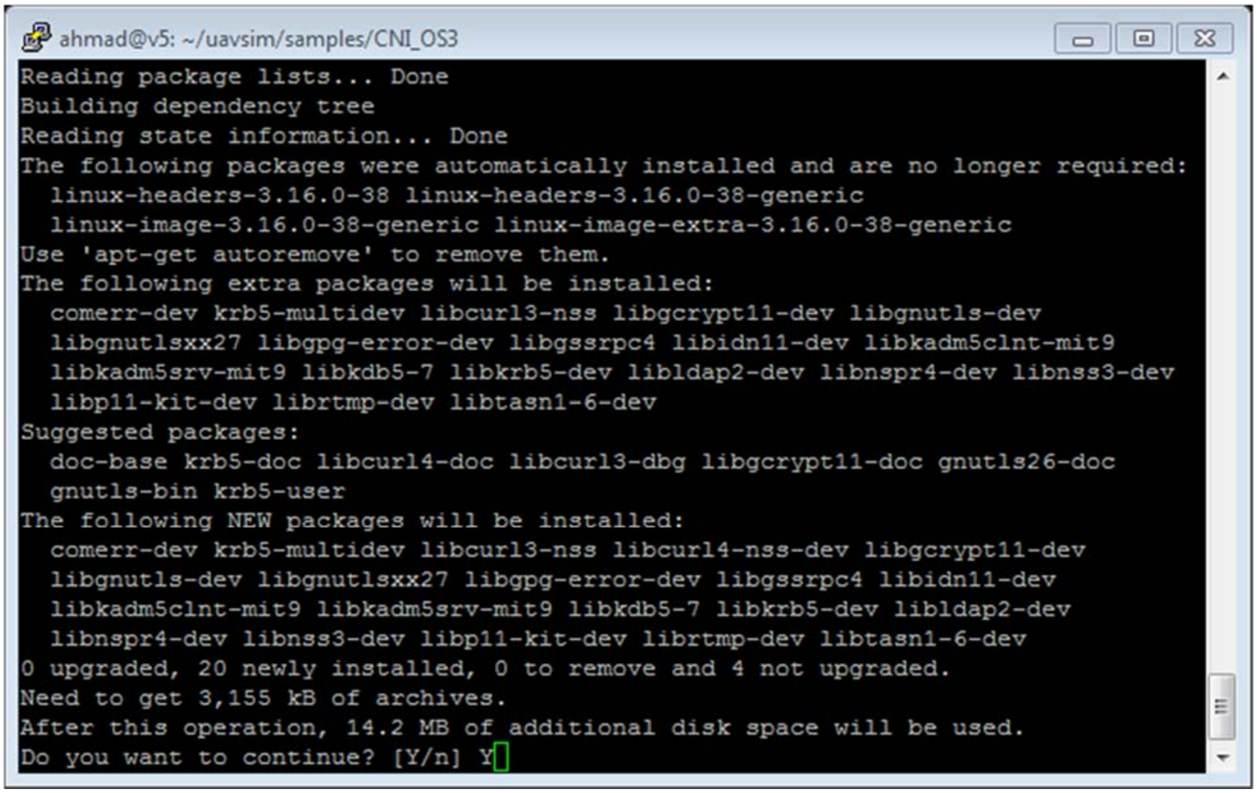


```
networklayer/rsvp_te/SignallingMsg_m.cc
networklayer/ted/LinkStatePacket_m.cc
networklayer/ted/TED_m.cc
networklayer/xmipv6/MobilityHeader_m.cc
transport/contract/SCTPCommand_m.cc
transport/contract/TCPCommand_m.cc
transport/contract/UDPControlInfo_m.cc
transport/rtp/reports_m.cc
transport/rtp/RTCPPacket1_m.cc
transport/rtp/RTCPPacket2_m.cc
transport/rtp/RTCPPacket3_m.cc
transport/rtp/RTPInnerPacket_m.cc
transport/rtp/RTPInterfacePacket_m.cc
transport/rtp/RTPPacket_m.cc
transport/rtp/RTPParticipantInfo_m.cc
transport/rtp/RTPSenderControlMessage_m.cc
transport/rtp/RTPSenderStatusMessage_m.cc
transport/rtp/profiles/avprofile/RTPMpegPacket_m.cc
transport/sctp/SCTPMessage_m.cc
transport/tcp_common/TCPSegment_m.cc
transport/udp/UDPPacket_m.cc
Creating shared library: ../out/gcc-debug//src/libinet.so
make[1]: Leaving directory `/home/ahmad/uavsim/samples/inet/src'
ahmad@v5:~/uavsim/samples/inet$
```

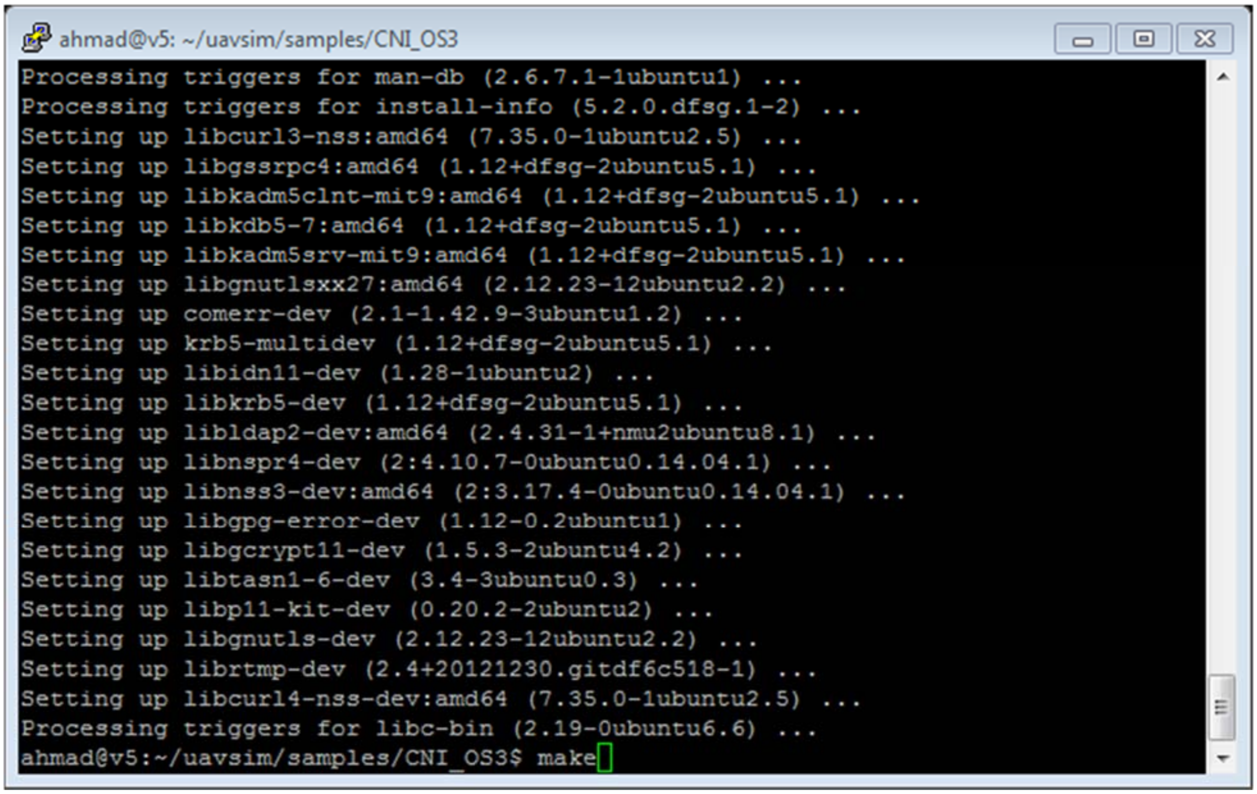
http://www-os3.kn.e-technik.tu-dortmund.de/index.php/downloads-1/viewdownload/7-help/2-installation-guide



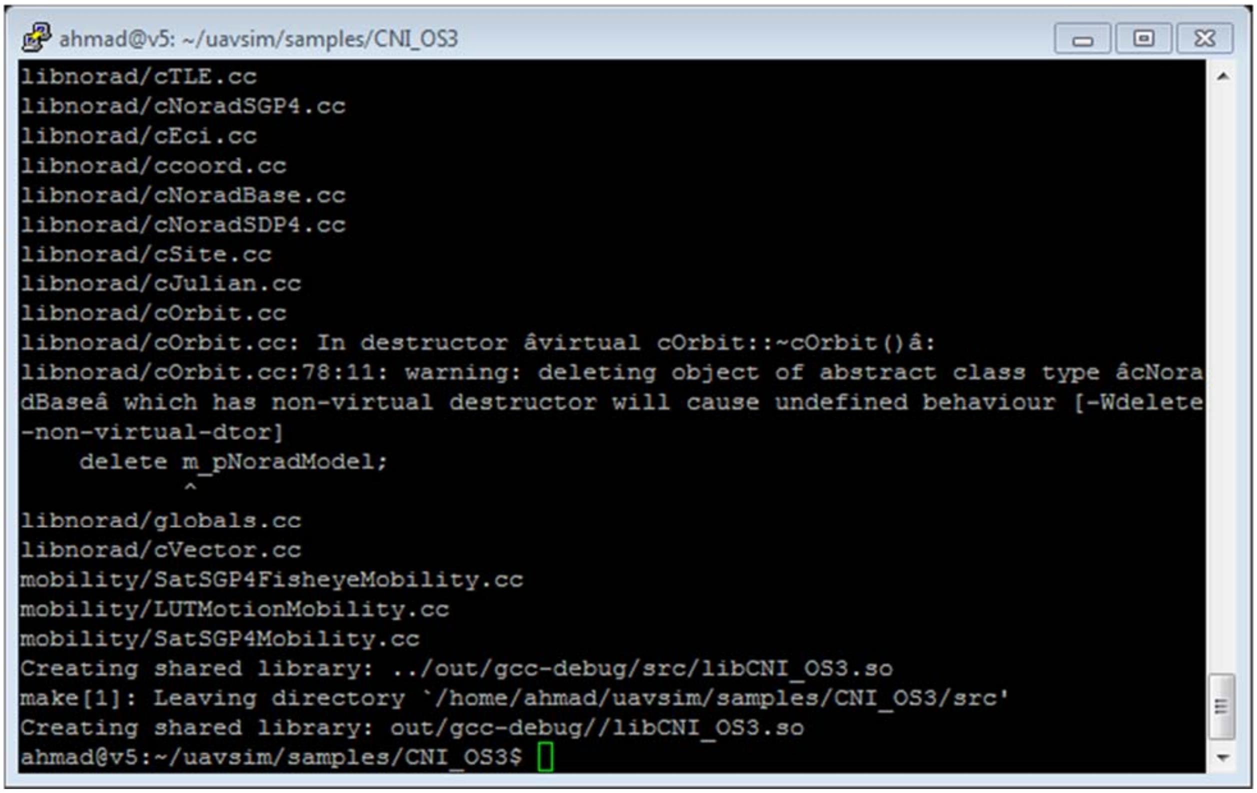
```
ahmad@v5:~/uavsim/samples/CNI_OS3$ sudo apt-get install libcurl4-nss-dev
```



```
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-3.16.0-38 linux-headers-3.16.0-38-generic
  linux-image-3.16.0-38-generic linux-image-extra-3.16.0-38-generic
Use 'apt-get autoremove' to remove them.
The following extra packages will be installed:
  comerr-dev krb5-multidev libcurl3-nss libgcrypt11-dev libgnutls-dev
  libgnutlsxx27 libgpg-error-dev libgssrpc4 libidn11-dev libkadm5clnt-mit9
  libkadm5srv-mit9 libkdb5-7 libkrb5-dev libldap2-dev libnspr4-dev libnss3-dev
  libp11-kit-dev librtmp-dev libtasn1-6-dev
Suggested packages:
  doc-base krb5-doc libcurl4-doc libcurl3-dbg libgcrypt11-doc gnutls26-doc
  gnutls-bin krb5-user
The following NEW packages will be installed:
  comerr-dev krb5-multidev libcurl3-nss libcurl4-nss-dev libgcrypt11-dev
  libgnutls-dev libgnutlsxx27 libgpg-error-dev libgssrpc4 libidn11-dev
  libkadm5clnt-mit9 libkadm5srv-mit9 libkdb5-7 libkrb5-dev libldap2-dev
  libnspr4-dev libnss3-dev libp11-kit-dev librtmp-dev libtasn1-6-dev
0 upgraded, 20 newly installed, 0 to remove and 4 not upgraded.
Need to get 3,155 kB of archives.
After this operation, 14.2 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```



```
Processing triggers for man-db (2.6.7.1-1ubuntu1) ...
Processing triggers for install-info (5.2.0.dfsg.1-2) ...
Setting up libcurl3-nss:amd64 (7.35.0-1ubuntu2.5) ...
Setting up libgssrpc4:amd64 (1.12+dfsg-2ubuntu5.1) ...
Setting up libkadm5clnt-mit9:amd64 (1.12+dfsg-2ubuntu5.1) ...
Setting up libkdb5-7:amd64 (1.12+dfsg-2ubuntu5.1) ...
Setting up libkadm5srv-mit9:amd64 (1.12+dfsg-2ubuntu5.1) ...
Setting up libgnutlsxx27:amd64 (2.12.23-12ubuntu2.2) ...
Setting up comerr-dev (2.1-1.42.9-3ubuntu1.2) ...
Setting up krb5-multidev (1.12+dfsg-2ubuntu5.1) ...
Setting up libidn11-dev (1.28-1ubuntu2) ...
Setting up libkrb5-dev (1.12+dfsg-2ubuntu5.1) ...
Setting up libldap2-dev:amd64 (2.4.31-1+nmu2ubuntu8.1) ...
Setting up libnspr4-dev (2:4.10.7-0ubuntu0.14.04.1) ...
Setting up libnss3-dev:amd64 (2:3.17.4-0ubuntu0.14.04.1) ...
Setting up libgpg-error-dev (1.12-0.2ubuntu1) ...
Setting up libgcrypt11-dev (1.5.3-2ubuntu4.2) ...
Setting up libtasn1-6-dev (3.4-3ubuntu0.3) ...
Setting up libp11-kit-dev (0.20.2-2ubuntu2) ...
Setting up libgnutls-dev (2.12.23-12ubuntu2.2) ...
Setting up librtmp-dev (2.4+20121230.gitdf6c518-1) ...
Setting up libcurl4-nss-dev:amd64 (7.35.0-1ubuntu2.5) ...
Processing triggers for libc-bin (2.19-0ubuntu6.6) ...
ahmad@v5:~/uavsim/samples/CNI_OS3$ make
```



```
libnorad/cTLE.cc
libnorad/cNoradSGP4.cc
libnorad/cEci.cc
libnorad/ccoord.cc
libnorad/cNoradBase.cc
libnorad/cNoradSDP4.cc
libnorad/cSite.cc
libnorad/cJulian.cc
libnorad/cOrbit.cc
libnorad/cOrbit.cc: In destructor âvirtual cOrbit::~cOrbit()â:
libnorad/cOrbit.cc:78:11: warning: deleting object of abstract class type âcNora
dBaseâ which has non-virtual destructor will cause undefined behaviour [-Wdelete
-non-virtual-dtor]
    delete m_pNoradModel;
           ^
libnorad/globals.cc
libnorad/cVector.cc
mobility/SatSGP4FisheyeMobility.cc
mobility/LUTMotionMobility.cc
mobility/SatSGP4Mobility.cc
Creating shared library: ../out/gcc-debug/src/libCNI_OS3.so
make[1]: Leaving directory `/home/ahmad/uavsim/samples/CNI_OS3/src'
Creating shared library: out/gcc-debug//libCNI_OS3.so
ahmad@v5:~/uavsim/samples/CNI_OS3$
```

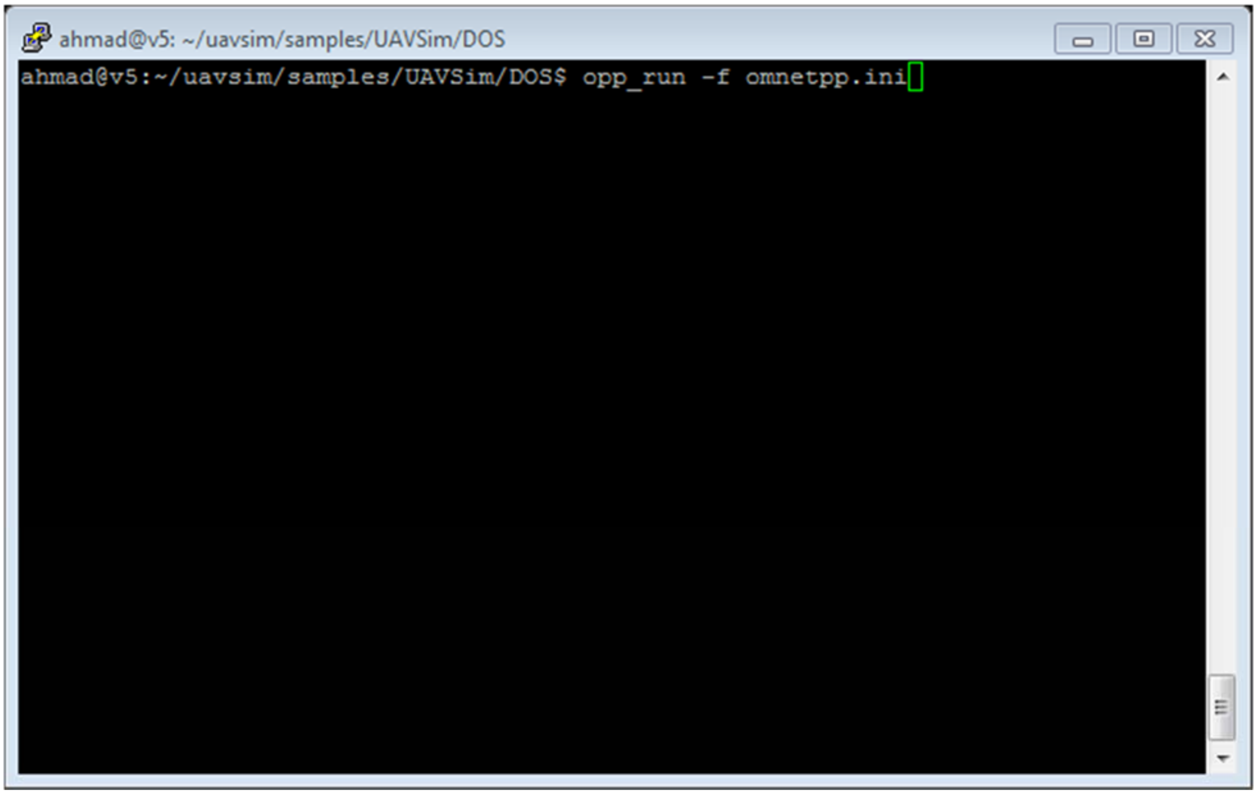
Now the UAVSim and all dependent module setup is complete.

7. Step 7. Running simulations in Command Line mode

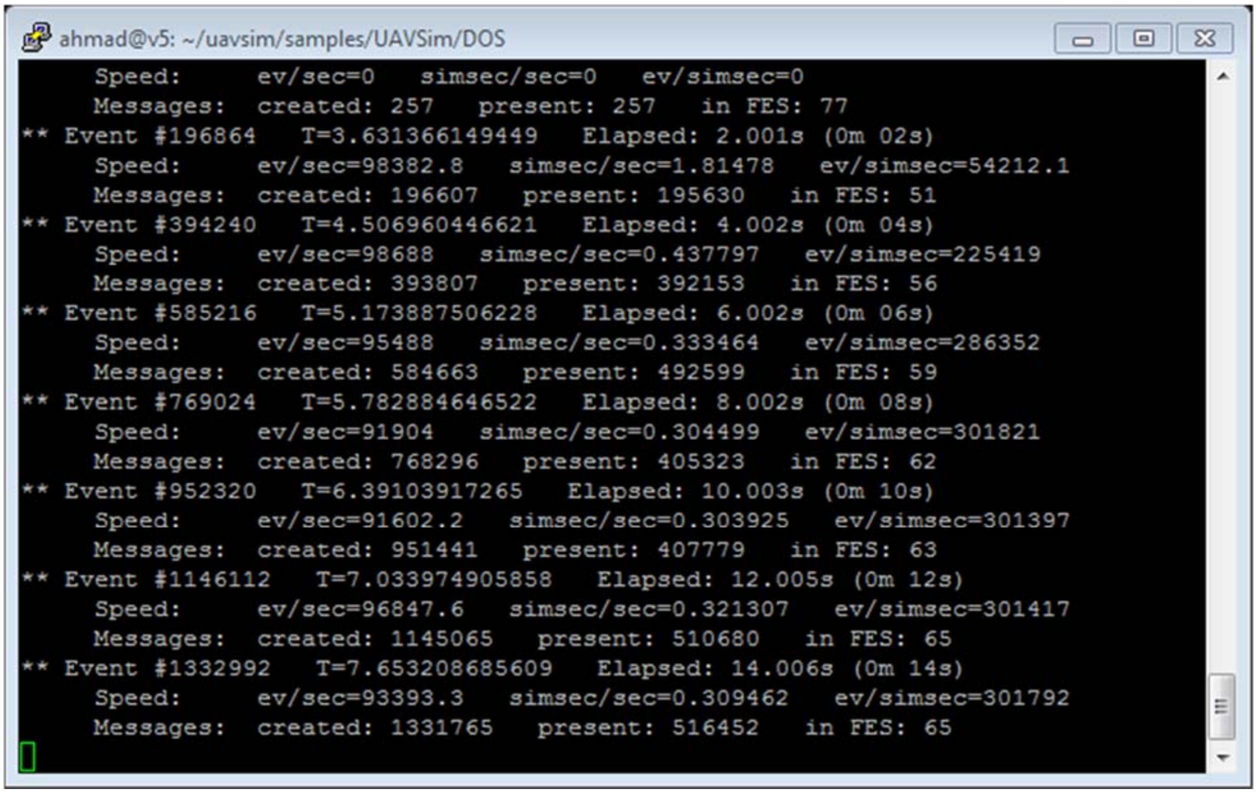
Here is a sample omnetpp.ini configuration file which indicates all parameters which need to be set before you can run a simulation in command line.

```
[General]
user-interface=Cmdenv

ned-path=/home/ahmad/uavsim/samples/UAVSim/:/home/ahmad/uavsim
/samples/UAVModels/:/home/ahmad/uavsim/samples/inet/src/:/home/ahmad/u
avsim/samples/inet/examples/
load-libs=/home/ahmad/uavsim/samples/inet/src/inet

#cmdenv-performance-display = true
cmdenv-express-mode = true
network = Net80211
#record-eventlog = true
#eventlog-message-detail-pattern = *:(not declaredOn(cMessage) and not
declaredOn(cNamedObject) and not declaredOn(cObject))

#fname-append-host = true
num-rngs = 3
sim-time-limit = 300s
**.mobility.rng-0 = 1
**.wlan[*].mac.rng-0 = 1
#debug-on-errors = true

#tkenv-plugin-path = ../../../etc/plugins

**.constraintAreaMinX = 0m
**.constraintAreaMinY = 0m
**.constraintAreaMinZ = 0m
**.constraintAreaMaxX = 200000m
**.constraintAreaMaxY = 200000m
**.constraintAreaMaxZ = 100000m
**.debug = true
**.coreDebug = false
**.host*.**.channelNumber = 0
**.attackhost*.**.channelNumber = 0

# channel physical parameters
*.channelControl.carrierFrequency = 4.5GHz
*.channelControl.pMax = 100.0W
*.channelControl.sat = -110dBm
*.channelControl.alpha = 2
*.channelControl.numChannels = 1

# mobility
**.host*.mobilityType = "LinearMobility"
**.host*.mobility.initFromDisplayString = false
```

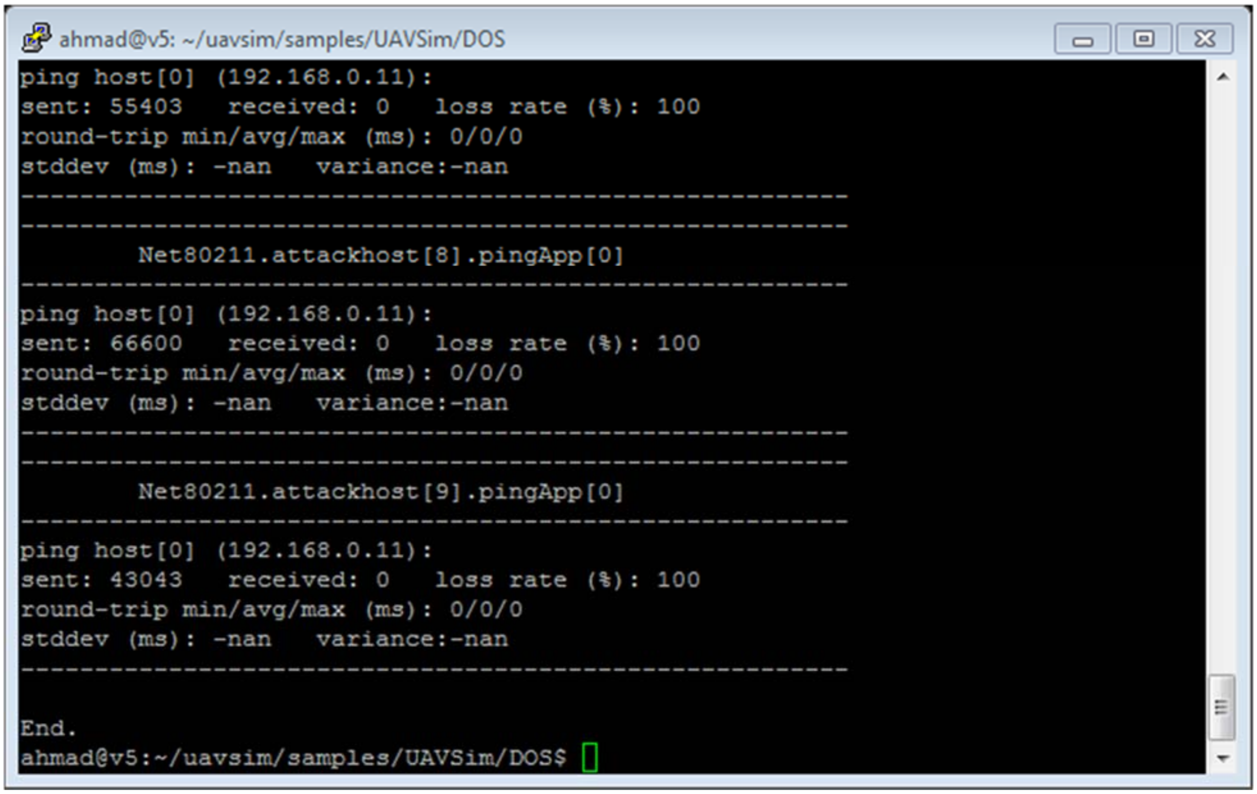
```
**.host*.mobility.changeInterval = truncnormal(2s, 0.5s)
**.host*.mobility.changeAngleBy = normal(0deg, 30deg)
**.host*.mobility.speed = truncnormal(100mps, 80mps)
**.host*.mobility.updateInterval = 10ms

# ping app
*.host[0].numPingApps = 0
*.host[1].numPingApps = 1
*.host[*].numPingApps = 2
*.attackhost[*].numPingApps = 1
*.host[*].pingApp[0].destAddr = "host[0]"
*.host[*].pingApp[1].destAddr = "host[1]"
*.attackhost[*].pingApp[0].destAddr = "host[0]"

**.pingApp[0].startTime = uniform(1s,5s)
*.attackhost[*].pingApp[0].startTime = uniform(0.01s,0.01s)
*.attackhost[*].pingApp[0].sendInterval = 0.0001s
**.pingApp[1].startTime = 2s+uniform(1s,5s)
**.pingApp[*].printPing = false

# nic settings
**.wlan[*].bitrate = 2Mbps
**.wlan[*].mgmt.frameCapacity = 10
**.wlan[*].mac.address = "auto"
**.wlan[*].mac.maxQueueSize = 14
**.wlan[*].mac.rtsThresholdBytes = 3000B
**.wlan[*].mac.retryLimit = 7
**.wlan[*].mac.cwMinData = 7
**.wlan[*].mac.cwMinBroadcast = 31

*.host[*].wlan[*].radio.transmitterPower = 50W
*.attackhost[*].wlan[*].radio.transmitterPower = 100W
**.wlan[*].radio.thermalNoise = -110dBm
**.wlan[*].radio.sensitivity = -85dBm
**.wlan[*].radio.pathLossAlpha = 2
**.wlan[*].radio.snirThreshold = 4dB
**.wlan[*].radio.radioModel = "NakagamiModel"

#description = "host1 pinging host0"
*.numHosts = 10
#${host=50,100,50,100}
*.numAttackHosts = 10
#${attackhost=0,0,0,0 ! host}
```



```
ahmad@v5:~/uavsim/samples/UAVSim/DOS$ opp_run -f omnetpp.ini
```



```
     Speed:     ev/sec=0   simsec/sec=0   ev/simsec=0
     Messages:  created: 257   present: 257   in FES: 77
** Event #196864   T=3.631366149449   Elapsed: 2.001s (0m 02s)
     Speed:     ev/sec=98382.8   simsec/sec=1.81478   ev/simsec=54212.1
     Messages:  created: 196607   present: 195630   in FES: 51
** Event #394240   T=4.506960446621   Elapsed: 4.002s (0m 04s)
     Speed:     ev/sec=98688   simsec/sec=0.437797   ev/simsec=225419
     Messages:  created: 393807   present: 392153   in FES: 56
** Event #585216   T=5.173887506228   Elapsed: 6.002s (0m 06s)
     Speed:     ev/sec=95488   simsec/sec=0.333464   ev/simsec=286352
     Messages:  created: 584663   present: 492599   in FES: 59
** Event #769024   T=5.782884646522   Elapsed: 8.002s (0m 08s)
     Speed:     ev/sec=91904   simsec/sec=0.304499   ev/simsec=301821
     Messages:  created: 768296   present: 405323   in FES: 62
** Event #952320   T=6.39103917265   Elapsed: 10.003s (0m 10s)
     Speed:     ev/sec=91602.2   simsec/sec=0.303925   ev/simsec=301397
     Messages:  created: 951441   present: 407779   in FES: 63
** Event #1146112   T=7.033974905858   Elapsed: 12.005s (0m 12s)
     Speed:     ev/sec=96847.6   simsec/sec=0.321307   ev/simsec=301417
     Messages:  created: 1145065   present: 510680   in FES: 65
** Event #1332992   T=7.653208685609   Elapsed: 14.006s (0m 14s)
     Speed:     ev/sec=93393.3   simsec/sec=0.309462   ev/simsec=301792
     Messages:  created: 1331765   present: 516452   in FES: 65
```



```
ping host[0] (192.168.0.11):
sent: 55403   received: 0   loss rate (%): 100
round-trip min/avg/max (ms): 0/0/0
stddev (ms): -nan   variance:-nan
--------------------------------------------------------
--------------------------------------------------------
        Net80211.attackhost[8].pingApp[0]
--------------------------------------------------------
ping host[0] (192.168.0.11):
sent: 66600   received: 0   loss rate (%): 100
round-trip min/avg/max (ms): 0/0/0
stddev (ms): -nan   variance:-nan
--------------------------------------------------------
--------------------------------------------------------
        Net80211.attackhost[9].pingApp[0]
--------------------------------------------------------
ping host[0] (192.168.0.11):
sent: 43043   received: 0   loss rate (%): 100
round-trip min/avg/max (ms): 0/0/0
stddev (ms): -nan   variance:-nan
--------------------------------------------------------


End.
ahmad@v5:~/uavsim/samples/UAVSim/DOS$
```

B. PART 2: Installation on a UNIX Desktop

1. Step 1: