

CORE Install Ubuntu 16.04

Install and update Ubuntu 16.04 LTS desktop (64-bit PC).

- If creating a VM you probably want 2+ CPUs, 2+GB RAM, 20GB or more HDD, depending on your host system.
- This was tested using VirtualBox 5.1.6 on Mac OS X 10.11.6 El Capitan. VirtualBox guest additions were also installed.

Install Latest Open Source CORE

Dependencies

Install dependency packages using apt-get:

```
sudo apt-get install bash bridge-utils ebttables iproute libtk-img python tcl8.5 tk8.5 autoconf
automake gcc libev-dev make pkg-config python-dev libreadline-dev imagemagick help2man
```

Install Quagga with MDR (for MANET routing):

```
wget http://downloads.pf.itd.nrl.navy.mil/ospf-manet/quagga-0.99.21mr2.2/quagga-mr_0.99.21mr2.2_amd64.
deb
sudo dpkg -i quagga-mr_0.99.21mr2.2_amd64.deb
```

Latest CORE Source from NRL (alternately, use Github)

This installs the latest CORE source from NRL Subversion (tested using file retrieved on 10/5/16):

```
wget http://downloads.pf.itd.nrl.navy.mil/core/source/nightly_snapshots/core-svnsnap.tgz
tar xzvf core-svnsnap-20160930.tgz
mv core core-20160930
cd core-20160930
./bootstrap.sh
./configure
make -j4
# this puts the GUI under /usr/local/lib/core
# Python module under /usr/local/lib/python2.7/dist-packages/
# binaries under /usr/local/sbin/core-daemon,vnoded,vcmd and /usr/local/bin/core-gui
sudo make install
```

Test Your CORE Install

In one terminal, start the backend using:

```
sudo core-daemon
```

In a second terminal, launch the GUI:

```
core-gui
```

Open and run the sample scenario:

- Choose *File > Open > sample1.imn*
- Press the start button (green arrow)
- Choose *Widgets > Adjacency > OSPFv2 >* and wait for blue lines (OSPF routing convergence)
- double-click **n11** (laptop in upper-left)
- Type in terminal: `ping 10.0.0.7`
- Drag e.g. **n7** in and out of range to start/stop pings.

If the above ping succeeds, your CORE install is good to go.

Terminal

CORE (44996 on ubuntu1604) sample1.imn

File Edit Canvas View Tools Widgets Session Help

new TCP connect GUI has connect ne' GU SE File Ed reMaking SE make[1 lo make[2 00 make[2 de make[2 IP make[1 de make[2 de make[2 SE make[2 Ch make[1 Ex jeffa sa jeffa SE jeffa j jeffa

```
root@n11:/tmp/pycore.44996/n11.conf
```

```
root@n11:/tmp/pycore.44996/n11.conf# ping 10.0.0.7
```

```
PING 10.0.0.7 (10.0.0.7) 56(84) bytes of data:
```

```
64 bytes from 10.0.0.7: icmp_seq=1 ttl=59 time=553 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=2 ttl=59 time=357 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=3 ttl=59 time=351 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=4 ttl=59 time=352 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=5 ttl=59 time=355 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=6 ttl=59 time=352 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=7 ttl=59 time=352 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=8 ttl=59 time=358 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=9 ttl=59 time=368 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=10 ttl=59 time=357 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=11 ttl=59 time=356 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=12 ttl=59 time=368 ms
```

```
64 bytes from 10.0.0.7: icmp_seq=13 ttl=59 time=351 ms
```

```
^C
```

wlan10 mobility script

Script file: sample1.scen

0

loop resolution: 50 ms

zoom 100% CPU 11% (10/12) 197M free