Software Improvement for E-Box

Martin Plantinga

September/October 2014

1 Install (L)UBUNTU on an USB drive

Follow the installation procedure stated in the folder "1. How to install (L)UBUNTU"

Steps to install Lubuntu on a USB drive:

- run the Lubuntu live cd
- open a terminal and type: `gparted`
- create 3 partitions:
  1. `BIOS/GRUB` of around 5 Mb
  2. `Swap area` of around 500 Mb
  3. `Ext4 Partition` and select `"/"` as mount point for the rest of the USB
- click **Install Lubuntu 14.10** on the desktop
- select **Something Else** to install in USB
- checkmark the `ext4` partition to format it
- before clicking install do **not** select a partition, but the generic path of the USB drive (*eg:* `/dev/sdc`). This has to be done for both where the partition table can be modified as where the device for boot loader installation is selected.
- click **Install**

2 TU/e WiFi Connection

3 TU/e VPN Connection

In order to install Matlab it is necessary to either be on the TU/e network or to connect to the VPN connection. In order to create a VPN connection in Lubuntu the `network-manager` package has to be added first:

```
»sudo apt-get update
»sudo apt-get install network-manager-pptp
```

This allows for the VPN dropdown to be available. After this has been installed follow the below procedure:

1. Go to the network icon and select `VPN Connections` → `Configure VPN` from the dropdown menu.
2. In the pop-up window click on Add and choose `Point-to-Point Tunneling Protocol (PPTP)` under `VPN` and click create
3. In the popup window change the following
   (a) Connection name: `TU/e VPN`
   (b) Gateway: `vpn.tue.nl`
   (c) Username: `tue\s******` (don't forget to add `tue\`)
   (d) Password: insert your password
4. Go to the `General` tab and check-mark `All users may connect to this network`
5. Click on Save

4 Matlab Installation

4.1 Local Installation

Local installation of Matlab saves the program on the local drive of the computer. The installation is as follows:

1. Make a new directory called software from the terminal:
   `# mkdir /software`
2. Install NFS client package:
   `# sudo apt-get install rpcbind nfs-common`
3. Mount the TU/e server to the created directory:
   `# mount hercules.tue.nl:/vol/hercules/software/linux /software`
4. Go to the directory where Matlab is located:
   `# cd /software/matlab-2014b`
5. Install the software:
   `# ./install`

4.2 Network Installation

1. Make a new directory in the `/usr` directory called `tuelocal`:
   `# /usr/tuelocal`
2. Add a NFS mount entry to `/etc/fstab`:
   (a) Install NFS client package:
   `# sudo apt-get install rpcbind nfs-common`
   (b) Open fstab file:
   `nano -w /etc/fstab`
   (c) Add a new line and exit:
   `# hercules.tue.nl:/vol/hercules/tuelocal/linux /usr/tuelocal nfs defaults 0 0`
3. Mount the previous entry:
   `# mount /usr/tuelocal`
4. Start Matlab as:
   `# /usr/tuelocal/matlab-2013b/bin/matlab`

4.3 Remove the following toolboxes for installation on USB

Aerospace Blockset	Image Acquisition
Aerospace Toolbox	Image Processing
Bioinformatics	Neural Network
Communication Systems	Parallel Computing
Computer Vision	SimBiology
Econometrics	SimElectronics
Financial Instruments	SimEvents
Financial Toolbox	SimHydraulics
Fuzzy Logic	SimMechanics
Global Optimization	SimPowerSystems
SimRF	Wavelet Toolbox

4.4 Add symbolic link

Before starting the installation process, add the following path for the symbolic link:

```
/home/ebox
/usr/local/bin
```

Create a symbolic link in `/usr/bin` such that Matlab will start from any directory

```
# cd /usr/bin
# ln -s /usr/local/matlab75/bin/./matlab /bin/matlab
```

4.5 Create Matlab Desktop Icon

Create empty file on Desktop named Matlab
`» touch /home/ebox/Desktop/Matlab`

Open file with gedit:
`» cd /home/ebox/Desktop`
`gedit Matlab`

Add the following entry:

```
[Desktop Entry]
X-AppInstall-Package=matlab-support
X-AppInstall-Popcon=10
X-AppInstall-Section=multiverse
Version=1.0
Type=Application
```

```
Terminal=false
Exec=/usr/local/MATLAB/R2013b/bin/matlab -desktop
Name=Matlab
Icon=/usr/share/app-install/icons/usr_share_icons_hicolor_48x48_apps_matlab.png
Categories=Development Math Science
Comment=Scientific computing environment
StartupNotify=true
StartupWMClass=com-mathworks-util-PostVMInit
X-Ubuntu-Gettext-Domain=app-install-data
```

NOTE: THIS HAS TO BE MODIFIED TO BE ABLE TO RUN AS ROOT!!

4.6 Running Matlab as Root

Go to a terminal and enter:

```
»sudo /usr/local/MATLAB/R2013b/bin/matlab
```

5 Installation of Additional Packages

5.1 Java

Sun Java 6 can be installed as follows:

```
» sudo add-apt-repository ppa:webupd8team/java
» sudo apt-get update
» sudo apt-get install oracle-java6-installer
```

5.2 GCC 4.7 & G++4.7 Compilers

Matlab 2013b is compatible up to versions 4.7.x. Install as follows:

```
»sudo add-apt-repository ppa:ubuntu-toolchain-r/test
» sudo apt-get update
» sudo apt-get install gcc-4.7 g++-4.7
```

Afterwards make a symbolic link:

```
»cd /usr/bin
» ln -sf gcc-4.7 gcc
» ln -sf g++-4.7 g++
```

5.3 Postgresql & Postgresql-client installation

Install `postgresql` and `postgresql-client`

```
# apt-get install postgresql
# apt-get install postgresql-client
```

5.4 Ethtool installation

Install `ethtool`

```
# apt-get install ethtool
```

5.5 Gedit

Gedit is very useful for modifying scripts.

```
# apt-get install gedit
```

5.6 SSH

SSH allows for remote access to computer

```
# apt-get install ssh
```

5.6.1 Remote access

From Mac terminal type: `ssh ebox@<ip address>`

5.7 Unetbootin

Unetbootin is used to make a bootable Live USB.

```
» add-apt-repository ppa:  gezakovacs/ppa
» apt-get update
» apt-get install unetbootin
```

5.8 Chromium Web Browser

```
» add-apt-repository ppa:  chromium-daily/stable
» apt-get update
» apt-get install chromium-browser
```

5.8.1 Flash for Chromium

```
» add-apt-repository ppa:  skunk/pepper-flash
» apt-get update
» apt-get install pepperflashplugin-installer
```

or

```
» apt-get install pepperflashplugin-nonfree
» update-pepperflashplugin-nonfree -install
```

6 Privileges

6.1 Change ownership of SVN directory

To change ownership to root ownership of the directory and its contents do as follows:

```
» chown -R root:root /home/ebox/Documents/SVN
```

Doublecheck ownership with the following command:

```
» ls -l
```

6.2 Change the execute permission to all files in the SVN directory

```
»chmod -R 755 /home/ebox/Documents/SVN
```

7 Modifications to SVN Scripts

7.1 ec_main.c

Path to this script is: `home/ebox/Documents/SVN/Targets/ectarget/ec_main.c`

Within the script in line 560 the path should be changed to:

```
u1ini("/usr/local/MATLAB/R2013b/rtw/c/ectarget/ectarget.u1");
```

8 Errors

8.1 make: command not found

Install the package `make`

```
» apt-get install make
```

8.2 cannot find -lpq

Install the package `libpq-dev`

```
» apt-get install libpq-dev
```

8.3 undefined reference to pthread_create

In the makefile (`SVN/Libs/soem/makefile`) change `-lptrhead` to `-pthread`.

In general, libraries should follow sources and objects on command line, and -lpthread is not an "option", it's a library specification. On a system with only libpthread.a installed,

8.4 collect2: error: ld returned 1 exit status

See prior error.

8.5 libengine.c:46:18: fatal error: /fir.h: no such file or directory

The error comes from `libengine.c`. In this script the libraries `fir.h, incomm.h` and `div.h` are included, the path however doesn't work and should be changed:

`#include </fir.h>` has to be `#include <../fir/fir.h>`
`#include </incomm.h>` has to be `#include <../incomm/incomm.h>`
`#include </div.h>` has to be `#include <../div/div.h>`

8.6 Escope v1.0.c:34:21: fatal error: gtk/gtk.h: No such file or directory

Install the library libgtk2.0-dev

```
apt-get install libgtk2.0-dev
```

Important to note that when installing this package both gcc as g++ are updated to version 4.9. In order to downgrade to version 4.7 do the following:

Remove gcc and g++: `apt-get purge gcc`

Add repository just in case:
`»sudo add-apt-repository ppa:ubuntu-toolchain-r/test`

Install gcc and g++:
```
» sudo apt-get update
» sudo apt-get install gcc-4.7 g++-4.7
```

Afterwards make a symbolic link:

```
»cd /usr/bin
» ln -sf gcc-4.7 gcc
» ln -sf g++-4.7 g++
```

8.7 gtk_graph.h:9:21: fatal_error: gtk/gtk.h: No such file or directory

See prior error.

8.8 sudo: ethtool: command not found

Install `ethtool`

```
# apt-get install ethtool
```

8.9 Incomm Client open failed

The link between `eseng.c` (*/home/ebox/svn/trunk/src/E-box/Apps/E-Scope/engine*) and the qs script is broken. Temporarily solution: run `eseng` from terminal when the ebox is connected.

```
# /home/ebox/svn/trunk/src/E-box/Apps/E-Scope/engine/eseng
```

Better solution:

- open `/home/ebox/svn/trunk/src/E-box/guiv25/qs.m`
- comment out lines 41 & 42: `cmd = ...` and `eval(cmd)`

- add the following line:
  `system('/home/ebox/svn/trunk/src/E-box/Apps/E-Scope/engine/eseng &');`
- save and exit

9 Error when connecting to another computer

When connecting to another computer `eseng` selects `eth0` by default. This can however be another port, i.e. `eth1`.

In order to fix this, `eseng` has to be called as follows:

```
# /home/ebox/svn/trunk/src/E-box/Apps/E-Scope/engine/eseng 4096 4096
0 ethx
```
where **x** stands for the correct port number
The `changeeth` script has to be run also.

10 Extra packages

10.1 Banner

```
# apt-get install sysvbanner
```

11 To Do

1. Create symbolic link for `geteth`, i.e.:

   ```
   # ln -s /home/ebox/svn/trunk/src/E-box/Scripts/geteth
   /bin/geteth
   ```

2. create `changeeth.m` global function
3. return eth port from `geteth` to `changeeth`

12 Notes

The paths within Matlab could already be added in order to prevent having to add them. Maybe a script can do this.