**Micronet Software Documentation**

**Introduction:**

This document outlines all of the software packages included in the Micronet installation images to provide a user with knowledge of the functionality included and to create a guide that would allow a user to build their own CR testbed with selected components from Micronet. It briefly discusses the purpose of each package and lists the commands needed to install the package via a terminal session. The configuration details for the Raspberry Pi are also included.

**Operating Systems:**

* Laptop – Ubuntu 12.04 LTS
* RaspberryPi – Raspbian

**Software Packages:**

Gnuradio:

This package offers a wide variety of DSP functions that can be easily implemented using gnuradio-companion, a GUI interface for gnuradio. It includes interfaces for common SDR hardware creating an extremely intuitive way to implement SDR applications. It can also be used as a platform for simulation of various radio concepts.

Installation: (Ubuntu)

Running the command below will download a script to install gnuradio along with all its dependencies, make it executable, and then run it. The source files for gnuradio will reside in the directory from which this command is run.

$ wget http://www.sbrac.org/files/build-gnuradio && chmod a+x ./build-gnuradio &&

./build-gnuradio

Installation: (Raspberry Pi)

Is there an easy way to get binaries as opposed to compiling?

Put script on site and list commands to download and run it

<http://k1gto.blogspot.com/2012/09/compiling-gnuradio-on-raspberry-pi-raspi.html>

RTL-SDR Driver:

Enables a device to communicate with the RTL SDR dongles.

Installation:

Pyrtlsdr:

Provides an easy to use python API for RTL SDR dongles.

Installation:

$ sudo apt-get install pyrtlsdr

Spectrum Analyzer Script:

This is a custom script that allows users to view any range of the radio spectrum that is covered by an RTL-SDR dongle.

Installation:

Add github for Micronet with this script.

RTL SDR Scanner:

This python script provides a spectrum analyzer with a GUI.

Installation:

<http://eartoearoak.com/software/rtlsdr-scanner/rtlsdr-scanner-installation>

XRDP:

This package makes both the Raspberry Pi and the laptop accessible using remote desktop protocol. This could prove useful for educational purposes if the user is not familiar with command line interfaces.

Installation:

$ sudo apt-get install XRDP

Cluster SSH:

This package allows a user to execute commands on multiple processors via SSH.

Installation:

$ sudo apt-get install clusterssh

Gedit:

Useful for editing text documents.

Installation:

$ sudo apt-get install gedit

FM Transmitter: (Raspberry Pi) Change to GRC Module once written

The Raspberry Pi can be easily used as a wideband FM transmitter. The only additional component required is a 16.4 cm wire connected to GPIO pin 4 to act as an antenna.

Installation:

$ wget http://omattos.com/pifm.tar.gz  
$ tar -zxvf Pifm.tar.gz

Usage:

$ sudo ./pifm wavfile.wav [freq] [sample rate]

kalibrate-rtl:

Allows for the frequency offset of the RTL to be compensated by analyzing local GSM transmissions.

Installation:

Other Dependencies Installed:

python-cheetah

libboost-all-dev

python-lxml

python-wxgtk2.8

python-numpy

python-lxml

libfftw3-dev

libsdl1.2-dev

python-scipy

python-matplotlib

python-tk

octave

liboctave-dev

libgsl0-dev

python-sphinx

libcppunit-dev

libuhd-dev

swig

python-qt4-dev

libqwt-dev

git

cmake

libusb-1.0-0-dev

**Raspberry Pi Configuration and Notes:**

Sometimes the raspberry pi may throw an error stating that the rtl driver is already in use and will not allow you to run a program, even if you are not currently running anything or have not run anything yet. In this case you just need to release the driver with:

$ sudo rmmod dvb\_usb\_rtl28xxu

Raspberry Pi’s can be configured headlessly by following the instructions in the following link.

<http://www.robertawood.com/blog/raspberry-pi/raspberry-pi-initial-setup-headless-no-monitor-or-keyboard-needed.html>

Access the configuration menu with:

$ sudo raspi-config

For our setup we want to maximize the processing power of the Pi by setting its clock rate to turbo (1000 MHz). It is also generally a good idea to expand the drive to occupy the entire SD card.

It will probably be useful to set a static IP address following these instructions:

<https://www.modmypi.com/blog/tutorial-how-to-give-your-raspberry-pi-a-static-ip-address>

Finally, it’s typically a good idea to get the latest firmware with:

$ sudo apt-get update

$ sudo apt-get upgrade

**Additional Software of interest:**

OpenBTS

OpenLTE

Liquid DSP

RedHawk