```
Last login: Fri Aug  1 09:42:47 on ttys003
You have mail.
mac02-3:VirtualBox VMs andrew$ pwd
/Users/andrew/VirtualBox VMs
mac02-3:VirtualBox VMs andrew$ cd dev
-bash: cd: dev: No such file or directory
mac02-3:VirtualBox VMs andrew$ cd
mac02-3:~ andrew$ cd dev
mac02-3:dev andrew$ cd python/qgisplugins/FarmSense
mac02-3:FarmSense andrew$ make deploy
pyuic4 -o ui_fseditfeature.py ui_fseditfeature.ui
#cd help; make html
echo >/dev/null
mkdir -p /Users/andrew/.qgis2/python/plugins/farmsense
rm -f /Users/andrew/.qgis2/python/plugins/farmsense/*.py*
rm -r -f /Users/andrew/.qgis2/python/plugins/farmsense/lib
cp -vf farmsense.py   fslayerconfig.py fsmqttclient.py fsutils.py  ui_fsaddfeature.py ui_fseditfeature.py fseditfeature.py fslay
er.py fslayermanager.py  fsaddfeature.py fsdevicemapdialog.py  __init__.py   /Users/andrew/.qgis2/python/plugins/farmsense
farmsense.py -> /Users/andrew/.qgis2/python/plugins/farmsense/farmsense.py
fslayerconfig.py -> /Users/andrew/.qgis2/python/plugins/farmsense/fslayerconfig.py
fsmqttclient.py -> /Users/andrew/.qgis2/python/plugins/farmsense/fsmqttclient.py
fsutils.py -> /Users/andrew/.qgis2/python/plugins/farmsense/fsutils.py
ui_fsaddfeature.py -> /Users/andrew/.qgis2/python/plugins/farmsense/ui_fsaddfeature.py
ui_fseditfeature.py -> /Users/andrew/.qgis2/python/plugins/farmsense/ui_fseditfeature.py
fseditfeature.py -> /Users/andrew/.qgis2/python/plugins/farmsense/fseditfeature.py
fslayer.py -> /Users/andrew/.qgis2/python/plugins/farmsense/fslayer.py
fslayermanager.py -> /Users/andrew/.qgis2/python/plugins/farmsense/fslayermanager.py
fsaddfeature.py -> /Users/andrew/.qgis2/python/plugins/farmsense/fsaddfeature.py
fsdevicemapdialog.py -> /Users/andrew/.qgis2/python/plugins/farmsense/fsdevicemapdialog.py
__init__.py -> /Users/andrew/.qgis2/python/plugins/farmsense/__init__.py
cp -vf ui_fsdevicemapdialog.py ui_fsconfigdialog.py  ui_fsaddfeature.py  ui_fseditfeature.py ui_fseditfeature.ui /Users/andrew/.
qgis2/python/plugins/farmsense
ui_fsdevicemapdialog.py -> /Users/andrew/.qgis2/python/plugins/farmsense/ui_fsdevicemapdialog.py
ui_fsconfigdialog.py -> /Users/andrew/.qgis2/python/plugins/farmsense/ui_fsconfigdialog.py
ui_fsaddfeature.py -> /Users/andrew/.qgis2/python/plugins/farmsense/ui_fsaddfeature.py
ui_fseditfeature.py -> /Users/andrew/.qgis2/python/plugins/farmsense/ui_fseditfeature.py
ui_fseditfeature.ui -> /Users/andrew/.qgis2/python/plugins/farmsense/ui_fseditfeature.ui
cp -vf resources_rc.py fsdevices.xml /Users/andrew/.qgis2/python/plugins/farmsense
resources_rc.py -> /Users/andrew/.qgis2/python/plugins/farmsense/resources_rc.py
fsdevices.xml -> /Users/andrew/.qgis2/python/plugins/farmsense/fsdevices.xml
cp -vf icon.png metadata.txt /Users/andrew/.qgis2/python/plugins/farmsense
icon.png -> /Users/andrew/.qgis2/python/plugins/farmsense/icon.png
metadata.txt -> /Users/andrew/.qgis2/python/plugins/farmsense/metadata.txt
cp -vfr i18n /Users/andrew/.qgis2/python/plugins/farmsense
i18n -> /Users/andrew/.qgis2/python/plugins/farmsense/i18n
cp -vfr lib /Users/andrew/.qgis2/python/plugins/farmsense
lib -> /Users/andrew/.qgis2/python/plugins/farmsense/lib
lib/__init__.py -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/__init__.py
lib/__init__.pyc -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/__init__.pyc
lib/fsdevicemaps.py -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/fsdevicemaps.py
lib/fsdevicemaps.pyc -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/fsdevicemaps.pyc
lib/fsdevicetypes.py -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/fsdevicetypes.py
lib/fsdevicetypes.pyc -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/fsdevicetypes.pyc
lib/fslogging.py -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/fslogging.py
lib/fslogging.pyc -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/fslogging.pyc
lib/fssettings.py -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/fssettings.py
lib/fssettings.pyc -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/fssettings.pyc
lib/mosquitto.py -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/mosquitto.py
lib/mosquitto.pyc -> /Users/andrew/.qgis2/python/plugins/farmsense/lib/mosquitto.pyc
#cp -vfr help/build/html /Users/andrew/.qgis2/python/plugins/farmsense/help
mac02-3:FarmSense andrew$ ssh redmine
Welcome to Redmine, TurnKey Linux 11.3 / Ubuntu 10.04 Lucid LTS

  System information (as of Fri Aug 01 18:48:31 2014)

    System load:  0.08               Memory usage:  38%
    Processes:    135                Swap usage:    2%
    Usage of /:   89.3% of 16.73GB   IP address for eth0:  203.6.246.29

  TKLBAM (Backup and Migration):  NOT INITIALIZED

    To initialize TKLBAM, run the "tklbam-init" command to link this
    system to your TurnKey Hub account. For details see the man page or
    go to:

        http://www.turnkeylinux.org/tklbam

You have mail.
Last login: Fri Aug  1 09:29:30 2014 from 115.188.145.145
andrew@redmine ~$ sudo cat /home/anarsoul/.ssh/authorized_keys
[sudo] password for andrew:
AAAAB3NzaC1yc2EAAAABIwAAAQEAvNwOaqzL16/OZ5hwa6KTitIWuQJYUpL2ydLOh0tXzFRQPvb3t9m7Qy/K/AU+RxAO7O+8fWz9D9k5SocK6ijgxUtOMeDJIxQE4HmU
iMqtLwweWB6sgPvgJ7UTsFkYF2tTe5tArzy0HWfTfkdDdvxLmdwnHxDU4IGQKfBAiLcdlj4yDWaBkLfJsgloHPR2AE91D1X/l3cHQOQM735+rEDGzAM2i/bbJLTpdmfu
MTXorwiSMD//CD53IuUIbBMEG3gceIIk5EQtOl0cs+N+33LRWk5 E+R3M5OrwaZNeBPmBxz41DjRWM1rdQd5zmiO182dzwpmizyFvnFh11F0JnKG2gw== anarsoul@a
narsoul-laptop
andrew@redmine ~$ exit
logout
Connection to redmine closed.
mac02-3:FarmSense andrew$ pwd
/Users/andrew/dev/python/qgisplugins/FarmSense
mac02-3:FarmSense andrew$ ls
API.txt                         fseditfeature.py                junk                            ui_fsdevicemapdialog.py
Makefile                        fseditfeature.pyc               lib                             ui_fsdevicemapdialog.pyc
README.html                     fslayer.py                      metadata.txt                    ui_fsdevicemapdialog.ui
```

```
README.txt                      fslayerconfig.py                resources.qrc                   ui_fseditfeature.py
__init__.py                     fslayermanager.py               resources_rc.py                 ui_fseditfeature.pyc
farmsense.komodoproject         fsmqttclient.py                 server                          ui_fseditfeature.ui
farmsense.py                    fsmqttclient.pyc                snippets.py                     x.qgs
fsaddfeature.py                 fsutils.py                      ui_fsaddfeature.py              x.qgs.mldata
fsaddfeature.pyc                fsutils.pyc                     ui_fsaddfeature.pyc             xx.qgs
fsdevicemapdialog.py            help                            ui_fsaddfeature.ui              xx.qgs.mldata
fsdevicemapdialog.pyc           i18n                            ui_fsconfigdialog.py
fsdevices.xml                   icon.png                        ui_fsconfigdialog.ui
mac02-3:FarmSense andrew$ cat server/
cat: server/: Is a directory
mac02-3:FarmSense andrew$ ls server/
API.txt                 fsd.db                  fsdconfig.pyc           fsdmqtt.pyc             mosquitto.pyc
db                      fsd.py~                 fsdevices.pyc           fsdtest.py              out.log
dbx                     fsdaemon-tests.py       fsdevices.xml           index.html              pybeautify.py
dtest.py                fsdaemon.py             fsdleveldb.py           index.html.1            test.py
fsd                     fsdaemon.pyc            fsdleveldb.pyc          lib                     test.ui~
fsd.cfg                 fsdconfig.py            fsdmqtt.py              mosquitto.py            x.py
mac02-3:FarmSense andrew$ rm x.py
mac02-3:FarmSense andrew$ pwd
/Users/andrew/dev/python/qgisplugins/FarmSense
mac02-3:FarmSense andrew$ cd server/
mac02-3:server andrew$ ls
API.txt                 fsd.db                  fsdconfig.pyc           fsdmqtt.pyc             mosquitto.pyc
db                      fsd.py~                 fsdevices.pyc           fsdtest.py              out.log
dbx                     fsdaemon-tests.py       fsdevices.xml           index.html              pybeautify.py
dtest.py                fsdaemon.py             fsdleveldb.py           index.html.1            test.py
fsd                     fsdaemon.pyc            fsdleveldb.pyc          lib                     test.ui~
fsd.cfg                 fsdconfig.py            fsdmqtt.py              mosquitto.py            x.py
mac02-3:server andrew$ rm test.ui~ test.py
mac02-3:server andrew$ pwd
/Users/andrew/dev/python/qgisplugins/FarmSense/server
mac02-3:server andrew$ ls
API.txt                 fsd.db                  fsdconfig.pyc           fsdmqtt.pyc             mosquitto.pyc
db                      fsd.py~                 fsdevices.pyc           fsdtest.py              out.log
dbx                     fsdaemon-tests.py       fsdevices.xml           index.html              pybeautify.py
dtest.py                fsdaemon.py             fsdleveldb.py           index.html.1            x.py
fsd                     fsdaemon.pyc            fsdleveldb.pyc          lib
fsd.cfg                 fsdconfig.py            fsdmqtt.py              mosquitto.py
mac02-3:server andrew$ cat dtest.py
#!/usr/bin/python

import sys
import time
import os
import atexit
from signal import SIGTERM
import logging

logging.basicConfig(filename='/dev/stdout',level=logging.DEBUG)

#logging.debug('This message should go to the log file')
#logging.info('So should this')
#logging.warning('And this, too')

class MyDaemon(object):
    """
        A generic daemon class.
        Usage: subclass the Daemon class and override the run() method
    """
    startmsg = "started with pid %s"

    def __init__(self, pidfile, stdin='/dev/null', stdout='/dev/null', stderr='/dev/null'):
            self.stdin = stdin
            self.stdout = stdout
            self.stderr = stderr
            self.pidfile = pidfile

    def daemonize(self):
        """
        do the UNIX double-fork magic, see Stevens' "Advanced
        Programming in the UNIX Environment" for details (ISBN 0201563177)
        http://www.erlenstar.demon.co.uk/unix/faq_2.html#SEC16
        """
        try:
                pid = os.fork()
                if pid > 0:
                        # exit first parent
                        sys.exit(0)
        except OSError, e:
                sys.stderr.write("fork #1 failed: %d (%s)\n" % (e.errno, e.strerror))
                sys.exit(1)

        # decouple from parent environment
        os.chdir(".")
        os.setsid()
        os.umask(0)

        # do second fork
        try:
            pid = os.fork()
            if pid > 0:
                # exit from second parent
                sys.exit(0)
```

```
        except OSError, e:
            sys.stderr.write("fork #2 failed: %d (%s)\n" % (e.errno, e.strerror))
            sys.exit(1)

        # redirect standard file descriptors
        si = file(self.stdin, 'r')
        so = file(self.stdout, 'a+')
        se = file(self.stderr, 'a+', 0)

        pid = str(os.getpid())

        sys.stderr.write("\n%s\n" % self.startmsg % pid)
        sys.stderr.flush()

        if self.pidfile:
            file(self.pidfile,'w+').write("%s\n" % pid)

        atexit.register(self.delpid)
        os.dup2(si.fileno(), sys.stdin.fileno())
        os.dup2(so.fileno(), sys.stdout.fileno())
        os.dup2(se.fileno(), sys.stderr.fileno())

    def delpid(self):
        os.remove(self.pidfile)

    def start(self):
        """
        Start the daemon
        """
        # Check for a pidfile to see if the daemon already runs
        try:
            pf = file(self.pidfile,'r')
            pid = int(pf.read().strip())
            pf.close()
        except IOError:
            pid = None

        if pid:
            message = "pidfile %s already exist. Daemon already running?\n"
            sys.stderr.write(message % self.pidfile)
            sys.exit(1)

        # Start the daemon
        self.daemonize()
        self.run()

    def stop(self):
        """
        Stop the daemon
        """
        # Get the pid from the pidfile
        try:
                pf = file(self.pidfile,'r')
                pid = int(pf.read().strip())
                pf.close()
        except IOError:
                pid = None

        if not pid:
            message = "pidfile %s does not exist. Daemon not running?\n"
            sys.stderr.write(message % self.pidfile)
            return # not an error in a restart

        # Try killing the daemon process
        try:
            while 1:
                os.kill(pid, SIGTERM)
                time.sleep(0.1)
        except OSError, err:
            err = str(err)
            if err.find("No such process") > 0:
                if os.path.exists(self.pidfile):
                    os.remove(self.pidfile)
                else:
                    print str(err)
                    sys.exit(1)

    def restart(self):
        """
        Restart the daemon
        """
        self.stop()
        time.sleep(5)
        self.start()

    def run(self):
        """
        You should override this method when you subclass Daemon.
        It will be called after the process has been
        daemonized by start() or restart().
        """

class TaskDaemon(MyDaemon):
    def run(self):
```

```
        while True:
            ''' this function below is called every 1 second '''
            try:
                do_my_task()
            except Exception,ex:
                print "Error in doing do_my_task(): %s" % (ex)

            ''' sleep for 1 second before doing stuff again '''
            time.sleep(1)

def do_my_task():
    logging.info("%s" % (time.time()))

def main():

    PIDFILE = '/tmp/my_daemon.pid'
    daemon = TaskDaemon(PIDFILE)
    daemon_name = "TaskDemon"

    if len(sys.argv) == 2:
        #=============================================================
        if 'start' == sys.argv[1]:
            try:
                daemon.start()
            except Exception,ex:
                print "%s start() error: %s" % (daemon_name,ex)
        #=============================================================
        elif 'stop' == sys.argv[1]:
            try:
                daemon.stop()
            except Exception,ex:
                print "%s stop() error: %s" % (daemon_name,ex)
        #=============================================================
        elif 'restart' == sys.argv[1]:
            try:
                daemon.restart()
            except Exception,ex:
                print "%s retart() error: %s" % (daemon_name,ex)
        #=============================================================
        else:
            print "Unknown command"
            sys.exit(2)
        #=============================================================
        sys.exit(0)
    else:
        #=============================================================
        print "usage: %s start|stop|restart" % sys.argv[0]
        sys.exit(2)
        #=============================================================

if __name__ == "__main__":
    main()
mac02-3:server andrew$ more fsd
#!/usr/bin/python

import sys
import time
import os
import traceback
import atexit
import argparse
from signal import SIGTERM
import logging as Log
from fsdaemon import FSDaemon
from fsdconfig import FSDConfig as Config
from fsdleveldb import   FSLevelDB as LevelDB
from lib.fsdevicetypes import *
from lib.fsdevicemaps import *
from fsdmqtt import *
from fsdmqtt import FSDMqttClient as MQTTClient
from random import randrange
from math import *
import json

basename = os.path.basename(sys.argv[0]).split('.')[0]

Log.basicConfig(filename='/tmp/' + basename + '.log',level=Log.DEBUG)

class FSDMqttClientSub(FSDMqttClient):

    def __init__(self,creator):
        self._creator = creator
        super(FSDMqttClientSub,self).__init__(creator,'leveldb')
        try:
            self.leveldb = LevelDB(creator.config.get("leveldb"))
            self.leveldb.testMe()
        except Exception as e:
            Log.debug('Failed to open database ' + str(e))
            exit(0)

    def onConnect(self,mosq, obj, rc):
        mosq.subscribe("$SYS/broker/uptime/#", 2)
        Log.info( 'Sub: onConnect '  + self.host)
        Log.info("rc: "+str(rc))
```

```
class FSDMqttArduinoSimulator(FSDMqttClient):

    def __init__(self,creator):
        self._creator = creator
        super(FSDMqttArduinoSimulator,self).__init__(creator,'arduino')


    def loop(self):
        r = int(randrange(2)+1)
        t = int(time.time())

        if (t % r == 0):
            Log.info('Random publish' )
            lowbyte = hex( (randrange(4)+1) * 1000).upper()
            topic = '/xbee/0xABCD/' + lowbyte + '/anin/raw/' + str(randrange(4))
            #topic = '/xbee/0xABCD/0xABCD/anou/raw/' + str(randrange(4))
            payload = json.dumps([str(t), str(randrange(1023))])
            self.mqttc.publish(topic,payload ,0,False)
        super(FSDMqttArduinoSimulator,self).loop()



class FSDXBeeProcess(FSDMqttClient):

    def __init__(self,creator):
        self._creator = creator
        try:
            self.leveldb = LevelDB(creator.config.get("leveldb"))
        except Exception as e:
            Log.debug('Failed to open database ' + str(e))
            exit(0)
        super(FSDXBeeProcess,self).__init__(creator,'xbees')


    def onConnect(self,mosq, obj, rc):
        mosq.subscribe("/xbee/+/+/+/raw/#", 0)
        Log.info( 'Sub: onConnect '  + self.host)
        Log.info("rc: "+str(rc))

    """ Handle raw messages
    If device not found, create empty device map (note we should publish a devicemaps updated topic!)

    TODO
    If found:
    Add raw value to history (if history applicable) if changed (check last value)

    If found AND Mapped, then:

        1. perform conversion function
        2. publish new result

    """

    def onMessage(self,mosq, obj, msg):
        Log.info(msg.topic+" "+str(msg.qos)+" "+str(msg.payload))
        try:
            parts = msg.topic.split("/")
            # xbee/hibyte/lowbyte/type/pin
            devicekey = '/xbee/' + parts[FSDeviceMap.HIBYTE] + '/' + parts[FSDeviceMap.LOBYTE] + '/' + parts[FSDeviceMap.DTYPE]
+ '/' + parts[FSDeviceMap.DPIN+1]
mac02-3:server andrew$ ls
API.txt                 fsd.db                  fsdconfig.pyc           fsdmqtt.pyc             mosquitto.pyc
db                      fsd.py~                 fsdevices.pyc           fsdtest.py              out.log
dbx                     fsdaemon-tests.py       fsdevices.xml           index.html              pybeautify.py
dtest.py                fsdaemon.py             fsdleveldb.py           index.html.1            x.py
fsd                     fsdaemon.pyc            fsdleveldb.pyc          lib
fsd.cfg                 fsdconfig.py            fsdmqtt.py              mosquitto.py
mac02-3:server andrew$ rm
mac02-3:server andrew$ rm *~
mac02-3:server andrew$ ls
API.txt                 fsd.db                  fsdevices.pyc           fsdtest.py              out.log
db                      fsdaemon-tests.py       fsdevices.xml           index.html              pybeautify.py
dbx                     fsdaemon.py             fsdleveldb.py           index.html.1            x.py
dtest.py                fsdaemon.pyc            fsdleveldb.pyc          lib
fsd                     fsdconfig.py            fsdmqtt.py              mosquitto.py
fsd.cfg                 fsdconfig.pyc           fsdmqtt.pyc             mosquitto.pyc
mac02-3:server andrew$ rm dtest.py
mac02-3:server andrew$ ls
API.txt                 fsdaemon-tests.py       fsdevices.xml           index.html              pybeautify.py
db                      fsdaemon.py             fsdleveldb.py           index.html.1            x.py
dbx                     fsdaemon.pyc            fsdleveldb.pyc          lib
fsd                     fsdconfig.py            fsdmqtt.py              mosquitto.py
fsd.cfg                 fsdconfig.pyc           fsdmqtt.pyc             mosquitto.pyc
fsd.db                  fsdevices.pyc           fsdtest.py              out.log
mac02-3:server andrew$ rm fsd.cfg
mac02-3:server andrew$ cat fsd.
mac02-3:server andrew$ ls
API.txt                 fsdaemon-tests.py       fsdevices.xml           index.html              pybeautify.py
db                      fsdaemon.py             fsdleveldb.py           index.html.1            x.py
dbx                     fsdaemon.pyc            fsdleveldb.pyc          lib
fsd                     fsdconfig.py            fsdmqtt.py              mosquitto.py
fsd.cfg                 fsdconfig.pyc           fsdmqtt.pyc             mosquitto.pyc
fsd.db                  fsdevices.pyc           fsdtest.py              out.log
```

```
mac02-3:server andrew$ mv pybeautify.py ..
mac02-3:server andrew$ rm x.py
mac02-3:server andrew$ rm index.html*
mac02-3:server andrew$ pwd
/Users/andrew/dev/python/qgisplugins/FarmSense/server
mac02-3:server andrew$ pwd
/Users/andrew/dev/python/qgisplugins/FarmSense/server
mac02-3:server andrew$ ls
API.txt                 fsd.db                  fsdconfig.pyc           fsdmqtt.py              mosquitto.pyc
db                      fsdaemon-tests.py       fsdevices.pyc           fsdmqtt.pyc             out.log
dbx                     fsdaemon.py             fsdevices.xml           fsdtest.py
fsd                     fsdaemon.pyc            fsdleveldb.py           lib
fsd.cfg                 fsdconfig.py            fsdleveldb.pyc          mosquitto.py
mac02-3:server andrew$ cd ..
mac02-3:FarmSense andrew$ pwd
/Users/andrew/dev/python/qgisplugins/FarmSense
mac02-3:FarmSense andrew$ pwd
/Users/andrew/dev/python/qgisplugins/FarmSense
mac02-3:FarmSense andrew$ tar -zcvf server.tgz server/
a server
a server/API.txt
a server/db
a server/dbx
a server/fsd
a server/fsd.cfg
a server/fsd.db
a server/fsdaemon-tests.py
a server/fsdaemon.py
a server/fsdaemon.pyc
a server/fsdconfig.py
a server/fsdconfig.pyc
a server/fsdevices.pyc
a server/fsdevices.xml
a server/fsdleveldb.py
a server/fsdleveldb.pyc
a server/fsdmqtt.py
a server/fsdmqtt.pyc
a server/fsdtest.py
a server/lib
a server/mosquitto.py
a server/mosquitto.pyc
a server/out.log
a server/fsd.db/000163.ldb
a server/fsd.db/000165.ldb
a server/fsd.db/000618.log
a server/fsd.db/000619.ldb
a server/fsd.db/CURRENT
a server/fsd.db/LOCK
a server/fsd.db/LOG
a server/fsd.db/LOG.old
a server/fsd.db/MANIFEST-000616
a server/dbx/000013.log
a server/dbx/CURRENT
a server/dbx/LOCK
a server/dbx/LOG
a server/dbx/LOG.old
a server/dbx/MANIFEST-000012
a server/db/000031.log
a server/db/CURRENT
a server/db/LOCK
a server/db/LOG
a server/db/LOG.old
a server/db/MANIFEST-000030
mac02-3:FarmSense andrew$ pwd
/Users/andrew/dev/python/qgisplugins/FarmSense
mac02-3:FarmSense andrew$ ls -ltr
total 928
drwxr-xr-x   2 andrew  staff     68  6 Jun 17:07 i18n
drwxr-xr-x   6 andrew  staff    204  6 Jun 17:07 help
-rw-r--r--   1 andrew  staff    192 12 Jun 11:36 farmsense.komodoproject
-rw-r--r--   1 andrew  staff    105 12 Jun 11:53 resources.qrc
-rw-r--r--   1 andrew  staff   1034 12 Jun 14:01 icon.png
-rw-r--r--   1 andrew  staff    997 17 Jun 09:51 README.txt
-rw-r--r--   1 andrew  staff   1425 17 Jun 09:51 README.html
-rw-r--r--   1 andrew  staff    276 19 Jun 10:24 API.txt
-rwxr-xr-x   1 andrew  staff   3841 20 Jun 15:21 pybeautify.py
-rw-r--r--   1 andrew  staff   7584 12 Jul 11:30 ui_fsdevicemapdialog.ui
-rw-r--r--   1 andrew  staff   1275 15 Jul 17:14 snippets.py
-rw-r--r--   1 andrew  staff   7998 16 Jul 11:30 ui_fsdevicemapdialog.py
-rw-r--r--   1 andrew  staff   5364 16 Jul 11:30 resources_rc.py
-rw-r--r--   1 andrew  staff   1201 16 Jul 11:30 fsutils.py
-rw-r--r--   1 andrew  staff   1304 16 Jul 11:30 __init__.py
-rw-r--r--   1 andrew  staff   6507 17 Jul 09:42 ui_fsdevicemapdialog.pyc
drwxr-xr-x   7 andrew  staff    238 19 Jul 14:47 junk
-rw-r--r--   1 andrew  staff   1245 22 Jul 19:25 xx.qgs.mldata
-rw-r--r--   1 andrew  staff  15284 22 Jul 19:25 xx.qgs
-rw-r--r--   1 andrew  staff   5878 24 Jul 19:53 farmsense.py
-rw-r--r--   1 andrew  staff  15678 25 Jul 02:27 ui_fsconfigdialog.ui
-rw-r--r--   1 andrew  staff  14706 25 Jul 02:33 ui_fsconfigdialog.py
-rw-r--r--   1 andrew  staff   1054 27 Jul 12:37 metadata.txt
-rw-r--r--   1 andrew  staff   1511 29 Jul 20:19 fsutils.pyc
-rw-r--r--   1 andrew  staff   3709 29 Jul 20:19 fsaddfeature.pyc
-rw-r--r--   1 andrew  staff   5234 30 Jul 11:08 ui_fsaddfeature.ui
-rw-r--r--   1 andrew  staff   6010 30 Jul 11:29 ui_fsaddfeature.py
```

```
-rw-r--r--   1 andrew  staff   5041 30 Jul 11:44 fsaddfeature.py
-rw-r--r--   1 andrew  staff  10723 30 Jul 12:28 fsmqttclient.py
-rw-r--r--   1 andrew  staff  11358 30 Jul 17:55 fslayer.py
-rw-r--r--   1 andrew  staff   5224 30 Jul 19:55 ui_fsaddfeature.pyc
-rw-r--r--   1 andrew  staff  13063 30 Jul 19:55 fsmqttclient.pyc
-rw-r--r--   1 andrew  staff   3138 30 Jul 21:37 x.qgs.mldata
-rw-r--r--   1 andrew  staff  46569 30 Jul 21:37 x.qgs
-rw-r--r--   1 andrew  staff  11983 31 Jul 07:52 fslayermanager.py
-rw-r--r--   1 andrew  staff   2057 31 Jul 09:28 fsdevices.xml
-rw-r--r--   1 andrew  staff  17744 31 Jul 09:29 fslayerconfig.py
-rw-r--r--   1 andrew  staff  16930 31 Jul 09:29 fsdevicemapdialog.py
-rw-r--r--   1 andrew  staff   4493 31 Jul 09:37 Makefile
drwxr-xr-x  14 andrew  staff    476 31 Jul 09:45 lib
-rw-r--r--   1 andrew  staff  16907 31 Jul 09:45 fsdevicemapdialog.pyc
-rw-r--r--   1 andrew  staff   2577 31 Jul 22:20 fseditfeature.py
-rw-r--r--   1 andrew  staff   3102  1 Aug 17:29 ui_fseditfeature.ui
-rw-r--r--   1 andrew  staff   3748  1 Aug 17:29 ui_fseditfeature.py
-rw-r--r--   1 andrew  staff   3795  1 Aug 17:30 ui_fseditfeature.pyc
-rw-r--r--   1 andrew  staff   3976  1 Aug 17:30 fseditfeature.pyc
drwxr-xr-x  24 andrew  staff    816  1 Aug 18:53 server
-rw-r--r--   1 andrew  staff  93201  1 Aug 18:59 server.tgz
mac02-3:FarmSense andrew$ cd
mac02-3:~ andrew$ cd ~/Downloads/
mac02-3:Downloads andrew$ ls linino.tar.gz
mac02-3:Downloads andrew$ sshping 169.254.7.227
-bash: sshping: command not found
mac02-3:Downloads andrew$ ping 169.254.7.227
PING 169.254.7.227 (169.254.7.227): 56 data bytes
64 bytes from 169.254.7.227: icmp_seq=0 ttl=64 time=35.959 ms
^C
--- 169.254.7.227 ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 35.959/35.959/35.959/0.000 ms
mac02-3:Downloads andrew$ ssh 169.254.7.227
The authenticity of host '169.254.7.227 (169.254.7.227)' can't be established.
RSA key fingerprint is 3c:ce:b3:75:6b:42:92:6c:f7:66:0f:e9:48:55:92:17.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '169.254.7.227' (RSA) to the list of known hosts.
^C
mac02-3:Downloads andrew$ ssh root@169.254.7.227
root@169.254.7.227's password:
Linux wrtdev 3.2.0-4-amd64 #1 SMP Debian 3.2.57-3+deb7u2 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Aug  1 19:02:14 2014
root@wrtdev:~# quit
-bash: quit: command not found
root@wrtdev:~# exit
logout
Connection to 169.254.7.227 closed.
mac02-3:Downloads andrew$ scp linino.tar.gz root@169.254.7.227:/tmp
root@169.254.7.227's password:
linino.tar.gz                                                                 100% 4495     4.4KB/s   00:00
mac02-3:Downloads andrew$ ssh root@169.254.7.227
root@169.254.7.227's password:
Linux wrtdev 3.2.0-4-amd64 #1 SMP Debian 3.2.57-3+deb7u2 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Aug  1 19:10:16 2014 from 192.168.0.105
root@wrtdev:~# cd ~andrew
root@wrtdev:/home/andrew# cd openwrt/
root@wrtdev:/home/andrew/openwrt# ls
bin          build_dir  Config.in  docs   feeds.conf.default  LICENSE   package  rules.mk  staging_dir  tmp        tools
BSDmakefile  config     dl         feeds  include             Makefile  README   scripts   target       toolchain
root@wrtdev:/home/andrew/openwrt# ls feeds
linino      luci.tmp          management.tmp     oldpackages.tmp  packages.tmp   routing.tmp      telephony.tmp
luci        management        oldpackages        packages         routing        telephony
luci.index  management.index  oldpackages.index  packages.index   routing.index  telephony.index
root@wrtdev:/home/andrew/openwrt# ls package/
base-files  boot  devel  firmware  kernel  libs  Makefile  network  system  utils
root@wrtdev:/home/andrew/openwrt# mv l^C
root@wrtdev:/home/andrew/openwrt# wd
-bash: wd: command not found
root@wrtdev:/home/andrew/openwrt# ls linono
ls: cannot access linono: No such file or directory
root@wrtdev:/home/andrew/openwrt# cd linino
-bash: cd: linino: No such file or directory
root@wrtdev:/home/andrew/openwrt# ls
bin          build_dir  Config.in  docs   feeds.conf.default  LICENSE   package  rules.mk  staging_dir  tmp        tools
BSDmakefile  config     dl         feeds  include             Makefile  README   scripts   target       toolchain
root@wrtdev:/home/andrew/openwrt# pwd
/home/andrew/openwrt
root@wrtdev:/home/andrew/openwrt# cd feeds
root@wrtdev:/home/andrew/openwrt/feeds# ls ../package/
```

```
base-files  boot  devel  firmware  kernel  libs  Makefile  network  system  utils
root@wrtdev:/home/andrew/openwrt/feeds# cd ..
root@wrtdev:/home/andrew/openwrt# pwd
/home/andrew/openwrt
root@wrtdev:/home/andrew/openwrt# ls package/
base-files  boot  devel  firmware  kernel  libs  Makefile  network  system  utils
root@wrtdev:/home/andrew/openwrt# cd feeds
root@wrtdev:/home/andrew/openwrt/feeds# ls packages
admin  devel  ipv6  lang  libs  LICENSE  mail  multimedia  net  README  sound  utils
root@wrtdev:/home/andrew/openwrt/feeds# pwd
/home/andrew/openwrt/feeds
root@wrtdev:/home/andrew/openwrt/feeds# cd linino/
root@wrtdev:/home/andrew/openwrt/feeds/linino# ls
leveldb  libsnappy  python-plyvel
root@wrtdev:/home/andrew/openwrt/feeds/linino# mv * ../packages
root@wrtdev:/home/andrew/openwrt/feeds/linino# cd ../packages
root@wrtdev:/home/andrew/openwrt/feeds/packages# ls
admin  devel  ipv6  lang  leveldb  libs  libsnappy  LICENSE  mail  multimedia  net  python-plyvel  README  sound  utils
root@wrtdev:/home/andrew/openwrt/feeds/packages# ls
admin  devel  ipv6  lang  leveldb  libs  libsnappy  LICENSE  mail  multimedia  net  python-plyvel  README  sound  utils
root@wrtdev:/home/andrew/openwrt/feeds/packages# make
make: *** No targets specified and no makefile found.  Stop.
root@wrtdev:/home/andrew/openwrt/feeds/packages# cd ..
root@wrtdev:/home/andrew/openwrt/feeds# ls
linino      luci.tmp          management.tmp     oldpackages.tmp  packages.tmp   routing.tmp      telephony.tmp
luci        management        oldpackages        packages         routing        telephony
luci.index  management.index  oldpackages.index  packages.index   routing.index  telephony.index
root@wrtdev:/home/andrew/openwrt/feeds# make
make: *** No targets specified and no makefile found.  Stop.
root@wrtdev:/home/andrew/openwrt/feeds# pwd
/home/andrew/openwrt/feeds
root@wrtdev:/home/andrew/openwrt/feeds# cd
root@wrtdev:~# cd root@wrtdev:/home/andrew/openwrt/feeds
-bash: cd: root@wrtdev:/home/andrew/openwrt/feeds: No such file or directory
root@wrtdev:~# cd /home/andrew/openwrt/feeds
root@wrtdev:/home/andrew/openwrt/feeds# ls
linino      luci.tmp          management.tmp     oldpackages.tmp  packages.tmp   routing.tmp      telephony.tmp
luci        management        oldpackages        packages         routing        telephony
luci.index  management.index  oldpackages.index  packages.index   routing.index  telephony.index
root@wrtdev:/home/andrew/openwrt/feeds# cd ..
root@wrtdev:/home/andrew/openwrt# l
-bash: l: command not found
root@wrtdev:/home/andrew/openwrt# ls
bin          build_dir  Config.in  docs   feeds.conf.default  LICENSE   package  rules.mk  staging_dir  tmp        tools
BSDmakefile  config     dl         feeds  include             Makefile  README   scripts   target       toolchain
root@wrtdev:/home/andrew/openwrt# cd feeds/packages
root@wrtdev:/home/andrew/openwrt/feeds/packages# ls
admin  devel  ipv6  lang  leveldb  libs  libsnappy  LICENSE  mail  multimedia  net  python-plyvel  README  sound  utils
root@wrtdev:/home/andrew/openwrt/feeds/packages# mv libsnappy leveldb libs
root@wrtdev:/home/andrew/openwrt/feeds/packages# mv python-plyvel lang/
root@wrtdev:/home/andrew/openwrt/feeds/packages# cd ../..
root@wrtdev:/home/andrew/openwrt# make menuconfig


*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.

root@wrtdev:/home/andrew/openwrt# make package/symlinks
 make[1] package/symlinks
 make[2] prepare-tmpinfo
root@wrtdev:/home/andrew/openwrt# make menuconfig
Collecting package info: done
configuration written to .config

*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.

root@wrtdev:/home/andrew/openwrt#
root@wrtdev:/home/andrew/openwrt#
```