Inhaltsverzeichnis

Setting up Python 3.3 with SocketCan Support in Ubuntu

From python 3.3 onwards support the SocketCan, but Ubuntu still needs python 2.7 internally.

To install a separate python version 3.3, do
(taken from http://askubuntu.com/questions/244544/how-to-install-python-3-3)

we need C compiler and other stuff to compile Python

```
sudo apt-get install build-essential
```

SQLite libs need to be installed in order for Python to have SQLite support.

```
sudo apt-get install libsqlite3-dev
sudo apt-get install sqlite3 # for the command-line client
sudo apt-get install bzip2 libbz2-dev
```

Download and compile Python:

```
wget http://python.org/ftp/python/3.3.0/Python-3.3.0.tar.bz2
tar jxf ./Python-3.3.0.tar.bz2
cd ./Python-3.3.0
./configure --prefix=/opt/python3.3
make && sudo make install
```

some nice touches:

```
mkdir ~/bin
ln -s /opt/python3.3/bin/python3 ~/bin/py33
```

Python 3.3 is now available for your user account as `~/bin/py33`

Adding YAML

```
wget http://pyyaml.org/download/pyyaml/PyYAML-3.10.tar.gz
tar xvzf PyYAML-3.10.tar.gz
cd PyYAML-3.10/
sudo ~/bin/py33 setup.py  install
```

Starting SocketCAN

Load the Kernel modules

```
sudo modprobe can
sudo modprobe can_raw
sudo modprobe can_bcm
sudo modprobe vcan
```

Then you can create a new virtual CAN-Device by

```
sudo ip link add dev vcan0 type vcan
sudo ip link set up vcan0
```

or with a real device with buadrate

```
sudo ip link set can0 type can bitrate 500000
sudo ip link set up can0
```

(for SLCAN based interfaces see https://elinux.org/Bringing_CAN_interface_up)

With

```
ip link show
```

you should see something like

```
7: vcan0: <NOARP,UP,LOWER_UP> mtu 16 qdisc noqueue state UNKNOWN
link/can
```

To generate some traffic, you would need some tools out of the SocketCAN repository:

```
svn checkout http://svn.berlios.de/svnroot/repos/socketcan/trunk
cd trunk/can-utils
make
```

Now you'll need something to send. Save that as `test.log`:

test.log

```
(0.100000) vcan0 5D1#0000
(0.200000) vcan0 271#0100
(0.300000) vcan0 289#72027000
(0.400000) vcan0 401#081100000000
```

This can now be played in an endless loop with

```
./canplayer -l i -I test.log
```

You can see the traffic with

```
can-utils$ ./candump vcan0
```

From:
http://www.koehlers.de/wiki/ - **Steffen Köhlers Online- Bastelbuch**

Permanent link:
http://www.koehlers.de/wiki/doku.php?id=pc:pythoncad

Last update: **2022/02/16 09:10**