sudo btmon & hcitool lescan

if doesn't manage to lescan try:

hciconfig hci0 down

hciconfig hci0 up

or

service bluetooth restart

service dbus restart

from http://stackoverflow.com/questions/22062037/hcitool-lescan-shows-i-o-error

This worked for me:

As you pointed out, first run the hcitool in the background: sudo hcitool lescan --duplicates Note the use of "duplicates" so we keep on logging the changing RSSI value of the BLE device

Create a script (tester.sh) and insert the following code:

#!/bin/bash

while read address

do

read RSSI

timestamp=`date`

echo "$timestamp,$address,$RSSI"

done

in the above we're basically waiting for to lines from stdin (that's the 'read' lines). The first line containing 'read' is the MAC address of the device, the second 'read' line is to get the RSSI value. I also inserted the timestamp just for a more comprehensive answer

Now we use bash pipes to feed in the information we need like so:

sudo hcidump -a | egrep 'RSSI|bdaddr' | cut -f 8 -d ' ' | ./tester.sh > /tmp/result.csv

All we're doing here is using HCIDUMP, then egrep to filter out the lines containing the device address and the RSSI. The output of the egrep command is prepended with a tab, so in the subsequent cut command we have to use the 8th field to get what we're after since i'm separating on ' '. Last we feed this into our script which processes the values and output them in csv format. I then just redirect the output into the csv file

To manipulate the RSSI value you just need to modify the tester.sh file as needed.

from <http://raspberrypi.stackexchange.com/questions/18292/manipulate-rssi-value>

If scanning stops after 2-3 lines make sure you typed “sudo hcitool lescan –**duplicates**”

To listen on the server:

nc -vv -l 0.0.0.0 1234

To send messages:

sudo hcidump -a | nc 10.0.0.5 1234