Technical Report - Connecting the Parrot Bebop Drone to a Router (Multiple Bebop's)

Nicolas Gallardo (nicolasgallardo@ymail.com)

Tobias Naegeli (tobias.naegeli@inf.ethz.ch)

Introduction:

The Parrot Bebop Drone is one of the UAV's used in the TECHLAV test bed but is not compatible with ROS or any other network configurations besides its default configuration. In this technical report, the latter is addressed and in the following sections the steps to reconfigure the Bebop's network settings are presented. This procedure is necessary in order to connect multiple Bebop's to the same network accesses point. Each bebop has a default IP that will conflict with other Bebop's on the same network, thus the goal of these procedures is to allow the Bebop's IP to be reassigned and then connected to the desired network.

Procedure:

The out of the box firmware blocks certain networking capabilities that are standard in most Linux distributions. The author believes this is due to Parrot's SkyController, and the fact that these tools would allow the user to replicate the functionallity of the SkyController without buying it. To fix this do the following:

1. Connect to the Bebop drone over usb using the android debug tool. Therefore, wait until the drone booted and connect the drone to your computer and press the on/off button 4 times. To see if you have a connection type ifconfig into your terminal, you should see a new network interface called: usb0.
   A good tool for file transfer is the android debugger (adb)
   a. Install adb with the command:
      >> sudo apt-get install android-tools-adb
   b. Connect to the drone:
      >> adb connect 192.168.43.1:9050
   c. Make the partionion writable:
      >> adb shell mount –o remount,rw /

2. Add the files from github to the drone using the in the same path i.e. lib or sbin:
   a. You can do this with adb push: e.g. copy the flie libi.so from your computer to the drone folder /lib/ type:
      >> adp push libi.so /lib/libiw.so

      Do this for all the following files:

      /lib/libiw.so.29
      /lib/libiw.so
      /sbin/ifrename
      /sbin/iwconfig
      /sbin/iwevent

/sbin/iwgetid
/sbin/iwlist
/sbin/iwpriv
/sbin/iwspy
/bin/wpa_cli
/bin/wpa_passphrase
/bin/wpa_supplicant

3. type:
   >> adb shell
   To switch to the bebop shell.
4. Navigate to the folders /lib, /bin, sbin/ and change the mode to be able to execute the files using

   >> chmod 777 <filename>

5. On your host computer run the script connect with the following arguments:

   >> . /connect "<essid>" -p "<password>" [-a <address>] [-d <droneip>

   With droneip the actual drone ip to use for telnet. This is the default usb connection. There fore 192.168.43.1. the address is the new ip of the drone e.g 192.168.1.xxx.

   When you run the script you should see something like that:

6. If you want to directly connect to your network, you can do this via the on/off button. Therefore, edit the shell script shortpress_3.sh on your host pc.
   Fill in the essid, password and ip you want to have.
   First we have to back up the original longpress_0.sh file. On the drone, navigate to the folder /bin/onoffbutton.

   If a shortpress_3.sh script is already existent, make a backup of it>
   >> cp shortpress_3.sh shortpress_3_backup.sh

   >> exit

   Push the file shortpress_3.sh
   e.g. >> push shortpress_3.sh /bin/onoffbutton/shortpress_3.sh
   access the adb console by typing >> adb shell
   and change the mod of the file just pushed by using again the command
   chmod 777 <filename>

7. You can test if everything works by restarting the drone, wait until if is finishing with booding and press the on/off button three consecutive times, the Bebop will give a beep feedback, the connection should be established after 10 to 30s and you should be able to ping the drone.

8. Follow the readme file in the ARDrone-wpa2 folder to switch the bebop to a secured network.

Conclusion:

Applying these procedures to each Bebop will allow each to be assigned a unique IP address and connected to a specified access point. Not only does this allow for the use of a powerful network router, i.e. Linksys but it also allows multiple drones to share the same network without conflicting IP addresses.