

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

Nicolas Gallardo (nicolasgallardo@ymail.com)

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 access 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 functionality of the SkyController without buying it. As a work around, revert back to an older firmware distribution and obtain the following files, use these steps:

1. Download Firmware version 1.98.10:
<https://translate.google.com/translate?hl=en&sl=it&u=https%3A%2F%2Fficus23blog.wordpress.com%2F2015%2F02%2F15%2Fdowngrade-del-firmware-del-bebop%2F&prev=search>

(The author used the USB option)

2. Follow the instructions on the above web page and apply the old firmware (1.98.10) to drone by using a micro USB to USB adapter connected to the USB containing the firmware. Restart the drone and it should apply the update
IMPORTANT NOTE: Change the version.txt file (located in the home directory) using vi text editor to 0.0.0 so that the drone thinks it is not up to date, otherwise the update will not be applied. This can be done by using a telnet connection to the drones default IP 192.168.42.1

For more info on reverting back to most recent firmware version visit:

http://www.parrot.com/usa/support/parrot-bebop-drone/?_ga=1.259028810.15536573.1456100153

3. Remove the following files to the local computer (google SCP Linux if you dont know how to do this):

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

/sbin/iwgetid
/sbin/iwlist
/sbin/iwpriv
/sbin/iwspy

Please see this site for more info on the above files: <https://community.parrot.com/t5/Bebop-Drone/Iwconfig-not-found/m-p/114795#M10060>

4. Add the files to the drone with the new firmware in the same path i.e. lib or/sbin (again, google if you are unsure of how to do this)
5. Use the command:

```
chmod 777 <filename>
```

to make all the files executable on the drone

A similar problem occurred for Parrots older drone, ARDrone 2.0, and we will use the work around ROS package in the following steps:

6. Download the ARdrone-wpa2 package

```
sudo apt-get install ros-indigo-ardrone-wpa2
```

7. cd to the ardrone-wpa2 package and copy the following files to the drones /bin folder and repeat step 5 to make each file executable:

```
wpa_cli  
wpa_passphase  
wpa_suppl
```

8. In the ARDrone-wpa2 folder go in to scripts and open the connect script in an editor of your liking (nano, geddit, etc.)

-change any reference to ath0 to eth0

-change the default ip from 192.168.42.1 to 192.168.1.xxx (this is on the DRONEIP= line), i.e. any other unique ip address that is available on the network router

9. 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.