

# NETGEAR Nighthawk (R7000) Router Setup Guide For AR-Drone Use (Single Drone)

Made by Or Hanoach for MSS at the University of the Witwatersrand

April 14, 2018

## Contents

<b>0</b>	<b>Introduction</b>	<b>2</b>
<b>1</b>	<b>Pre-Setup</b>	<b>2</b>
1.1	Reset NetGear router to factory defaults . . . . .	2
<b>2</b>	<b>Router Configuration</b>	<b>2</b>
2.1	Enter router configuration via web browser . . . . .	2
2.2	Configure Wireless Settings . . . . .	3
2.3	Configure LAN Settings . . . . .	5
<b>3</b>	<b>Drone Configuration</b>	<b>6</b>
3.1	Changing Drone Settings . . . . .	6
3.2	Run wifi.sh from PC: . . . . .	7
<b>4</b>	<b>Run the ROS Driver</b>	<b>7</b>

## 0 Introduction

AR-Drones have their own internal routers. In order to send commands to an AR-Drone (eg. via ROS) you will need to connect your computer to the drone's wifi network. In order to connect to multiple drones you will need to have your router create a local LAN distributing its own IP address using DHCP. You will also need to reconfigure each drone's network settings to have it connect to the new LAN.

In the robotics room at MSL we use the NetGear Nighthawk (R7000), and thus this setup will guide you through configuring this specific router.

This guide was mostly taken from:

[https://github.com/AutonomyLab/ardrone\\_autonomy/wiki/Multiple-AR-Drones](https://github.com/AutonomyLab/ardrone_autonomy/wiki/Multiple-AR-Drones)

### **Note:**

There is a separate guide available for configuring your router and drones to use a single drone. This guide will also work for a single drone, but is more complicated. This over-complication is not needed if you only intend to use a single drone.

## 1 Pre-Setup

1. Connect the NetGear router to power and an ethernet cable from the PC to one of the numbered ports (not the internet port) on the back of the router.

### 1.1 Reset NetGear router to factory defaults

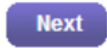
1. Take a thin screwdriver or anything else and press the "reset" button on the back of the router, until the lights turn off except for 2 lights (approx. 7 seconds)
2. Wait for the router to restart itself and the power light appears white.

## 2 Router Configuration

### 2.1 Enter router configuration via web browser

1. Open your favorite web browser.
  - (a) If you opened Google Chrome or Microsoft Edge re-evaluate your life, close the browser, and open a nice, open-source, web browser.
2. Make sure your proxy settings are off!
3. Go to <http://www.routerlogin.net/>
4. After the router finished searching for stuff, choose:  
☒ No, I want to configure the Internet connection myself.

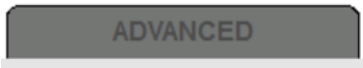
and:



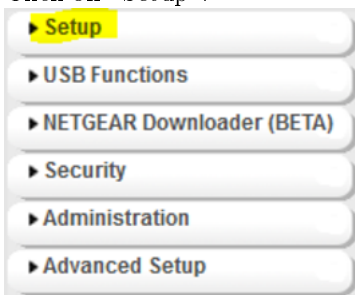
5. Default Username: admin  
Default Password: password

## 2.2 Configure Wireless Settings

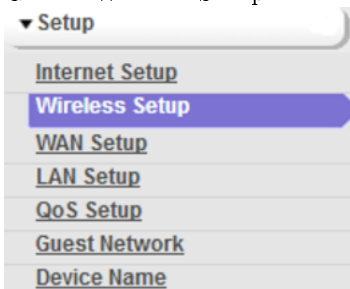
1. Go to the "ADVANCED" tab:



2. Click on "Setup":



3. Go to "Wireless Setup":



- (a) Set "Region" to "Africa (may not be necessary):

**Region Selection**

Region:

- (b) Under "Wireless Network (2.4GHz b/g/n)" and "Wireless Network (5GHz a/n/ac)" do the following: (5GHz maybe not necessary, but do it their to stay on the safe side):

- i. Change "Name (SSID)" to an SSID of your choice (in this guide we choose "dronenet", remember this because it will be relavent in the future)

**Name (SSID):**

- ii. In "Security Options" choose "None":

**Security Options**

- ☒ None  
☐ WPA2-PSK [AES]  
☐ WPA-PSK [TKIP] + WPA2-PSK [AES]  
☐ WPA/WPA2 Enterprise

- (c) Everything else should stay as is, so the final result should look like:

---

**Region Selection**  
Region: Africa ▼

---

☐ Enable Smart Connect - Let the router intelligently select the best WiFi band 2.4 GHz or 5 GHz for your WiFi connections. Smart Connect requires the 2.4 GHz and 5 GHz networks to use the same WiFi network name (SSID), security options and password.

---

**Wireless Network (2.4GHz b/g/n)**  
☒ Enable SSID Broadcast  
☒ Enable 20/40 MHz Coexistence  
Name (SSID): dronenet  
Channel: Auto ▼  
Mode: Up to 600 Mbps ▼  
Transmit Power Control: 100% ▼

---

**Security Options**  
☒ None  
☐ WPA2-PSK [AES]  
☐ WPA-PSK [TKIP] + WPA2-PSK [AES]  
☐ WPA/WPA2 Enterprise

---

**Wireless Network (5GHz a/n/ac)**  
☒ Enable SSID Broadcast  
Name (SSID): dronenet  
Channel: 44 ▼  
Mode: Up to 289 Mbps ▼  
Transmit Power Control: 100% ▼

---

**Security Options**  
☒ None  
☐ WPA2-PSK [AES]  
☐ WPA-PSK [TKIP] + WPA2-PSK [AES]  
☐ WPA/WPA2 Enterprise

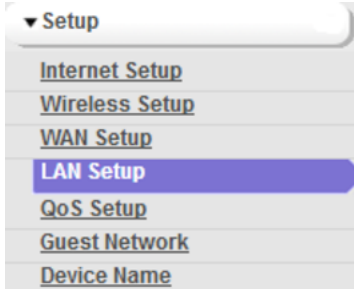
---

- (d) Click "Apply" at the top:

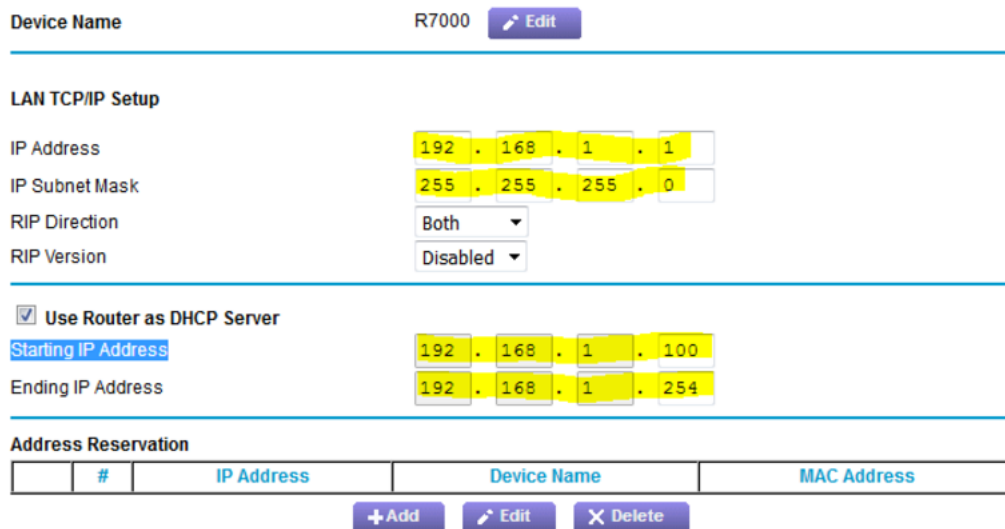
Apply ►

## 2.3 Configure LAN Settings

1. Go to "LAN Setup" on the left:



- (a) Make sure that the following is set:
  - i. "IP Address" is set to "192.168.1.1"
  - ii. "IP Subnet Mask" is set to "255.255.255.0"
  - iii. "RIP Direction" and "RIP Version" stay as they were.
  - iv. "Use Router as DHCP Server" is ticked and:
    - A. "Starting IP Address" is set to "192.168.1.100"
    - B. "Ending IP Address" is set to "192.168.1.254"
- (b) By the end it should look something like this:

A screenshot of the "LAN TCP/IP Setup" configuration page. At the top, the "Device Name" is "R7000" with an "Edit" button. The "LAN TCP/IP Setup" section contains: "IP Address" set to 192.168.1.1, "IP Subnet Mask" set to 255.255.255.0, "RIP Direction" set to "Both", and "RIP Version" set to "Disabled". Below this, the "Use Router as DHCP Server" checkbox is checked. The "Starting IP Address" is set to 192.168.1.100 and the "Ending IP Address" is set to 192.168.1.254. At the bottom, there is an "Address Reservation" table with columns for #, IP Address, Device Name, and MAC Address. Below the table are buttons for "+ Add", "Edit", and "X Delete".

#	IP Address	Device Name	MAC Address
---	------------	-------------	-------------

2. Click "Apply" at the top:



### 3 Drone Configuration

**Note:**

This section requires you to have a separate computer with wifi.  
This is due to your router not being configured to bridge the connection.

#### 3.1 Changing Drone Settings

1. Connect to your AR-Drone's ad-hoc wifi (directly to the drones wifi router, like you would connect to any wifi connection)
2. Open the terminal and telnet to the drone:

(a) Run:

```
telnet 192.168.1.1
```

3. create a file called wifi.sh in /data of the drone:

```
vi /data/wifi.sh
```

4. Copy the following into the file:

```
killall udhcpd
ifconfig ath0 down
iwconfig ath0 mode managed essid dronenet
ifconfig ath0 192.168.1.10 netmask 255.255.255.0 up
```

**Notice:**

- In line 3 change "dronenet" to the SSID you chose when setting up the router.
- The IP in line 4 should be unique to each drone, so change it between different drones.

**If you are not familiar with vi:**

- To enter insert mode in vi press "i".
- To save and exit vi press "esc" to exit insert mode and then ":wq")

5. Make the newly created file executable:

```
chmod +x /data/wifi.sh
```

6. Close the telnet connection:

```
exit
```

### 3.2 Run wifi.sh from PC:

**Notice:**

While the previous sections should only be done once, this step should be done whenever you want to make the AR-Drone connect to your wireless (for example every time you turn on the drone)

1. Connect to your AR-Drone's ad-hoc wifi (directly to the drones wifi router, like you would connect to any wifi connection)
2. Open a terminal
3. Execute the following command to remotely run wifi.sh on the drone:

```
echo "./data/wifi.sh" | telnet 192.168.1.1
```

- (a) Your computer should now disconnect from the drones network. This is due to the drone's network configurations being changed.
4. On the computer that is connected to the previously configured routers wireless network check that you have connection to the drone using ping:

```
ping 192.168.1.10
```

- (a) Where the IP address is the same as the one configured in wifi.sh of this drone.

## 4 Run the ROS Driver

Congratulations, everything should be working now!

When connecting to the drone using ROS you will need to make sure to supply the correct drone IP to ardrone\_autonomy. This can be done by:

1. If running ardrone\_autonomy directly in the command line use:

```
roslaunch ardrone_autonomy ardrone_driver -ip 192.168.1.10
```

2. If using a launch file add/change in the node line the argument:

```
args="-ip 192.168.1.10"
```

Thus you will end up with something like:

```
<node name="ardrone_driver" pkg="ardrone_autonomy"
type="ardrone_driver" args="-ip 192.168.1.10" />
```

**Notice:**

- The IP address in the above examples should be the same as the one configured in the wifi.sh file of this drone.
- Make sure to give the different nodes for different drones different names, i.e:

```
<node name="ardrone_driver1" pkg="ardrone_autonomy"  
type="ardrone_driver" args="-ip 192.168.1.10" />
```

and

```
<node name="ardrone_driver2" pkg="ardrone_autonomy"  
type="ardrone_driver" args="-ip 192.168.1.12" />
```

If using two drones, one with the ip 192.168.1.10 called “ardrone\_driver1” and the other with ip 192.168.1.12 called “ardrone\_driver2”

3. Program to your hearts content!