



Developer Study Guide

An introduction to Bluetooth® Beacons – Raspberry Pi

Release: 1.3.2

Document Version : 1.3.2

Last Updated: 21st December 2020

Running time: 30 to 60 minutes

Contents

CONTENTS.....	2
OVERVIEW.....	4
Beacons and AltBeacons	4
Prerequisites	4
Running Time	4
EXERCISE 1 – SET UP RASPBERRY PI3.....	5
Task 1 – Set up your Raspberry Pi3 with NOOBS	5
Task 2 – Install BlueZ onto the Raspberry Pi3	5
Exercise Complete	6
EXERCISE 2 – DEPLOY THE ALTBEACON PROFILE	7
Task 3 – Install the AltBeacon shell script on the Raspberry Pi3	7
Exercise Complete	8

Revision History

Version	Date	Author	Changes
1.0.0	16 th December 2016		Initial version
1.1.0	1st August 2018	Kai Ren Bluetooth SIG	<p>Raspberry Pi Tutorial: removed all the BDS references in the Raspberry Pi Tutorial.</p> <p>Raspberry Pi Tutorial: user doesn't need to download shell file on the host computer, user can just download the shell file by command on R Pi3 through SSH remote access or local access.</p> <p>Android Tutorial: add a "template" Android project for user, they can start copy and paste on it. User doesn't need to create a new Android Studio project.</p> <p>Android Tutorial: add a "fullSolution" Android project for user, user can build and install the App on their Android devices. This project is for user to debug if user has any problem for the implementation on "template" project.</p> <p>Android Tutorial: update Android project gradle, make the project can be compiled by latest Android Studio.</p> <p>Android Tutorial: double checked, the Android projects can be built on Windows and macOS host computer.</p> <p>iOS Tutorial: update some APIs which are deprecated.</p> <p>iOS Tutorial: optimize the work flow.</p>
1.2.0	18 th December 2018	Kai Ren Bluetooth SIG	<p>Name changed to "Develoepr Study Guide"</p> <p>Fixed an Android Studio compilation warning.</p>
1.3.0	11 th October 2019	Kai Ren Bluetooth SIG	<p>iOS</p> <p>retrofit the whole study guide to make it be compatible with Swift, instead of Objective-C.</p> <p>add new template Swift source code</p> <p>Raspberry Pi:</p> <p>Add latest Raspberry Pi4 support</p> <p>Update BlueZ installation from v5.49 to v5.50</p>
1.3.1	28 th October 2020	Martin Woolley Bluetooth SIG	Updated Android tutorial to incorporate permissions requests and be compatible with Android 10.
1.3.2	21 st December 2020	Martin Woolley Bluetooth SIG	Language changes

Overview

In this guide, we will show you how use a Raspberry Pi3 and to create your own AltBeacon – a Bluetooth® beacon – to use for later exercises with Android and iOS.

Beacons and AltBeacons

Beacons are low-power Bluetooth devices that advertise a data packet containing some unique identifier and secondary bits of information. The [AltBeacon](#) specification is an open and interoperable specification for proximity beacons using Bluetooth Low Energy (LE). AltBeacons, by design, are not tied to any particular manufacturer or chipset.

Prerequisites

- Keyboard and mouse
- Monitor with HDMI input
- HDMI cable
- Powered USB hub if you need more than 2 USB ports
- 8GB minimum SD card
- Power source for the Raspberry Pi3 (a USB to micro-USB cable)

Running Time

The two exercises should take 0.5 ~ 1 hours to complete, counting from the when you have all hardware in hand.

Exercise 1 – Set up Raspberry Pi3

This exercise assumes you have a brand new Raspberry Pi3, or want to guarantee success with the tutorial by installing a clean OS. If you have an OS setup you are happy with, move to step 2.

Task 1 – Set up your Raspberry Pi3 with NOOBS

1. Connect the Raspberry Pi3 to reliable USB power source (@5V, 2A), HDMI video out to a TV or monitor, and a USB keyboard and mouse.
2. Follow <https://www.raspberrypi.org/documentation/installation/> to setup your board, selecting Raspbian as the OS to ensure compatibility with the tutorial.

Task 2 – Install BlueZ onto the Raspberry Pi3

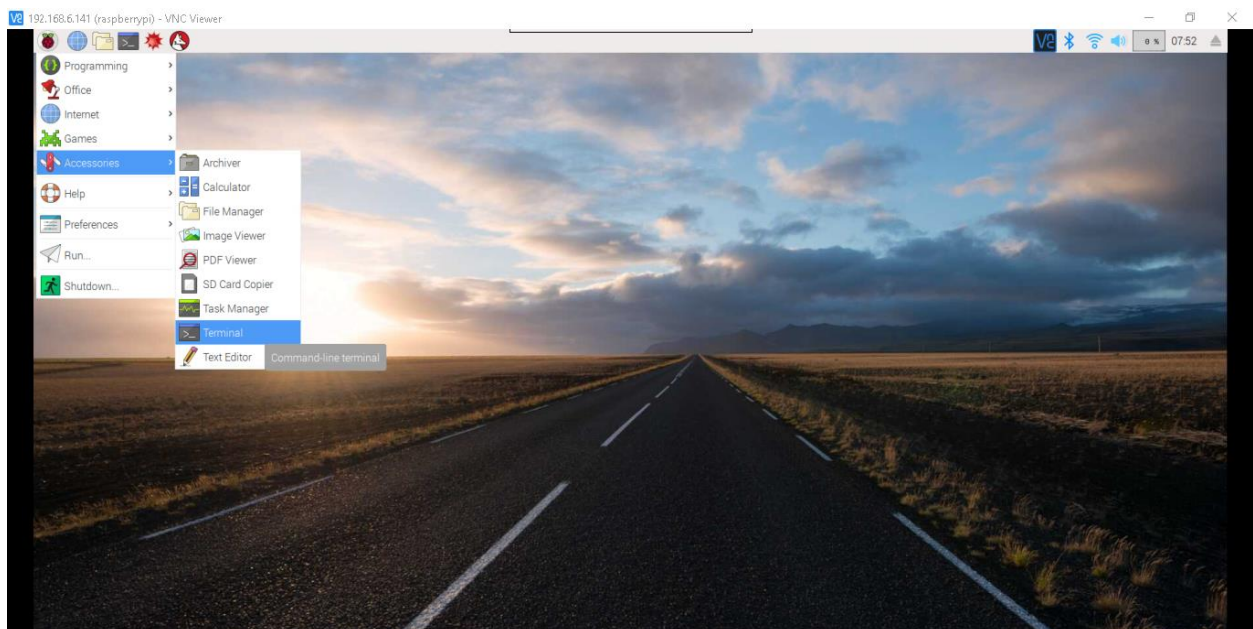
BlueZ is an open source Bluetooth stack for Linux. You will need it to allow your Raspberry Pi3 to act as a Bluetooth beacon. In the tutorial, we use [BlueZ v5.49](#) as the target version.

Internet Connection Needed: You will need to connect the Raspberry Pi3 to the Internet, either via Ethernet or Wi-Fi, in order to download the various packages required in this step. There are many sources of information for how to connect your Raspberry Pi3 to the Internet, and so this is beyond the scope of this guide.

Open the terminal on the Raspberry Pi3

- If you want to use Graphical User Interface (GUI) on Raspberry Pi3 locally, please do the following:

Open menu -> Accessories -> Terminal



- If you want to use remote access by SSH, please do the following:
<https://www.raspberrypi.org/documentation/remote-access/ssh/README.md>

Install prerequisites

Run the following command to install all the libraries required by BlueZ:

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install -y git bc libusb-dev libdbus-1-dev libglib2.0-dev libudev-dev libical-dev libreadline-dev
autoconf
```

Get BlueZ v5.49 source code

Run the following commands in the terminal to download BlueZ:

```
cd ~
wget http://www.kernel.org/pub/linux/bluetooth/bluez-5.49.tar.xz
tar -xvf bluez-5.49.tar.xz
cd bluez-5.49/
```

Run the following commands in the terminal to unzip and compile bluez (and please note that each one is a separate command):

```
sudo ./configure --prefix=/usr --mandir=/usr/share/man --sysconfdir=/etc --localstatedir=/var
make
sudo make install
```

This will take a while, so you may want to start on the Android or iOS labs in the meantime, or go for a walk! For supplementary details on installing BlueZ on a Raspberry Pi3, read the elinux.org page [here](#).

Verify the installation

Check the installation worked by running the following:

```
hciconfig
```

You should now see something like:

```
pi@raspberrypi:~ $ hciconfig
hci0:  Type: Primary  Bus: UART
      BD Address: B8:27:EB:7F:F3:F6  ACL MTU: 1021:8  SCO MTU: 64:1
      UP RUNNING
      RX bytes:829 acl:0 sco:0 events:58 errors:0
      TX bytes:4240 acl:0 sco:0 commands:58 errors:0
```

Exercise Complete

Congratulations! You have put a Bluetooth stack onto your Raspberry Pi3 and set it up for our next exercise: making an AltBeacon.

Exercise 2 – Deploy the AltBeacon profile

In this exercise, you will deploy AltBeacon shell file on Raspberry Pi3 as a starting point for an AltBeacon. You will then use this profile to generate a script that will set up the AltBeacon application on your Raspberry Pi3.

Task 3 – Install the AltBeacon shell script on the Raspberry Pi3

Now we will install the shell script onto the Raspberry Pi3 so that each time the Raspberry Pi3 is started, the AltBeacon will start broadcasting. This effectively turns the Raspberry Pi3 into a Beacon.

```
cd ~
wget https://www.bluetooth.com/~media/Files/developer/beacon\_shell.zip
unzip beacon_shell.zip
```

Try and run the file:

```
sudo sh ./beacon.sh
```

If it worked, you should see output such as:

```
hci0:  Type: Primary  Bus: UART
      BD Address: B8:27:EB:7F:F3:F6  ACL MTU: 1021:8  SCO MTU: 64:1
      UP RUNNING
      RX bytes:2277 acl:0 sco:0 events:144 errors:0
      TX bytes:5828 acl:0 sco:0 commands:144 errors:0

< HCI Command: ogf 0x08, ocf 0x0008, plen 15
  1F 02 01 1A 1B FF 18 01 BE AC 00 00 00 00 01
> HCI Event: 0x0e plen 4
  01 08 20 00
```

Add the script to the the bootup

Open the built-in vi editor to edit the bootup script:

```
sudo vi /home/pi/.bashrc
```

move the cursor to the bottom line and type **o** (switch vi to insert a line after bottom line), add the following line:

```
sudo sh /home/pi/beacon.sh
```

Press **ESC** on your keyboard (switch back to command mode) and save the file by following command:

```
:wq
```

Restart the Raspberry Pi3 using the command:

```
sudo reboot
```

When the Raspberry Pi3 has rebooted, verify the beacon deployment. Open the terminal by locally or remote access, if beacon is deployed successfully, you will see below output such as:

```
Linux raspberrypi 4.14.49-v7+ #1 SMP Fri Jun 15 03:10:12 UTC 2018 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Jun 29 23:15:13 2018
hci0: Type: Primary Bus: UART
      BD Address: B8:27:EB:7F:F3:F6 ACL MTU: 1021:8 SCO MTU: 64:1
      UP RUNNING
      RX bytes:2277 acl:0 sco:0 events:144 errors:0
      TX bytes:5828 acl:0 sco:0 commands:144 errors:0

< HCI Command: ogf 0x08, ocf 0x0008, plen 15
  1F 02 01 1A 1B FF 18 01 BE AC 00 00 00 00 01
> HCI Event: 0x0e plen 4
  01 08 20 00
```

Exercise Complete

Congratulations! You have successfully created a real AltBeacon using a Raspberry Pi3. Next, choose the Android or iOS labs to create an app that will see your AltBeacon.