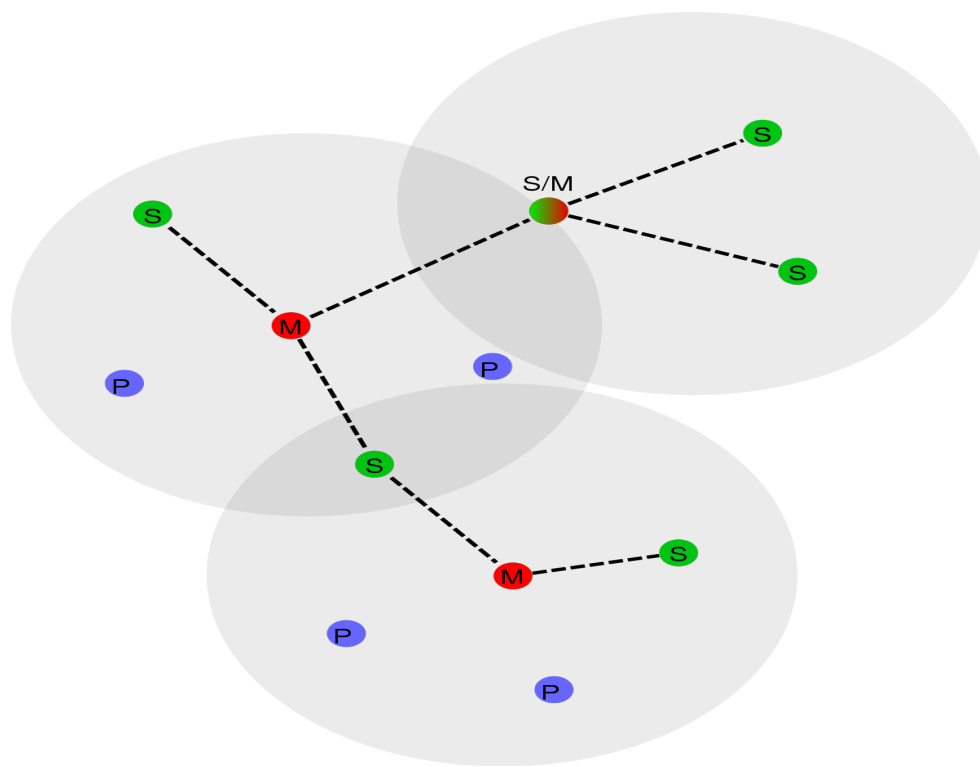


Especialización del Teleinformática

Análisis de conectividad (BLUETOOTH Low Energy)

Parámetros Anunciate y Escáneres



Daniel Rico Bonilla

Junio 2020.

Universidad Francisco Distrital José de Caldas.

Especialización de Teleinformática.

Asignatura de Teletráfico.

Abstract

This Bluetooth specification defines fundamental requirements to enable an interoperable mesh networking solution for Bluetooth low energy wireless technology.

Tabla de Contenidos

Capítulo 1

Introducción e información general

1.1. Arquitectura.

1.2. Pila Protocolos.

1.3. Detección de Estados

1.4. Evento advertising

Capítulo 2

2.1. Objetivo

2.2. Desarrollo

2.3. Implementación

2.3. Configuración

2.4 Configuración Intervalo de escaneo

2.5. Configuración de paquetes Advertising

Capítulo 3

3.1. Discusión

3.2. Resultados

3.3. Conclusiones

Lista de referencias

Capítulo 1

Introducción e información general

Bluetooth Low Energy (BLE) es una tecnología desarrollado por Bluetooth Special Interest Group (**SIG**). Se ha diseñado como una tecnología complementaria a Bluetooth clásico para garantizar un consumo de energía bajo, y menor tiempo de establecimiento de conexión. A pesar del uso de la misma banda de frecuencia y las similitudes compartidas, BLE debe considerarse un nuevo estándar con objetivos y aplicaciones diferentes. BLE está diseñado para la transmisión de pequeñas cantidades de datos (tiempos de transmisión muy pequeños) y por lo tanto de ultra-bajo consumo de energía. No está pensado para mantener una conexión entre dispositivos por un largo tiempo transmitiendo grandes cantidades de datos a alta velocidad. Esto permite que los dispositivos están activos sólo cuando se les pide la transmisión de datos.

BLE reduce notablemente la potencia de 15 mW y la potencia media a 5 μ W del Bluetooth clásico. Además realiza una gestión del consumo de energía en modo espera, utiliza menos canales RF para la comunicación entre dispositivos e implementa protocolos de comunicación simples, alcanzando eficiencias energéticas que pueden llegar a ser 20 veces mayores que las de Bluetooth clásico.

1.1. Arquitectura.

La topología de red de BLE es de tipo estrella. Los dispositivos Master pueden tener varias conexiones de capa de enlace con periféricos (Slaves) y simultáneamente realizar búsquedas de otros dispositivos. Por otro lado un dispositivo en rol de esclavo solo puede tener una conexión de capa de enlace con un único Master. Además, un dispositivo puede

enviar datos en modo Broadcast, eventos de Advertising, sin esperar ninguna conexión; esto permite enviar datos a los dispositivos en estado Scanning sin necesidad de establecer la conexión Master-Slave.

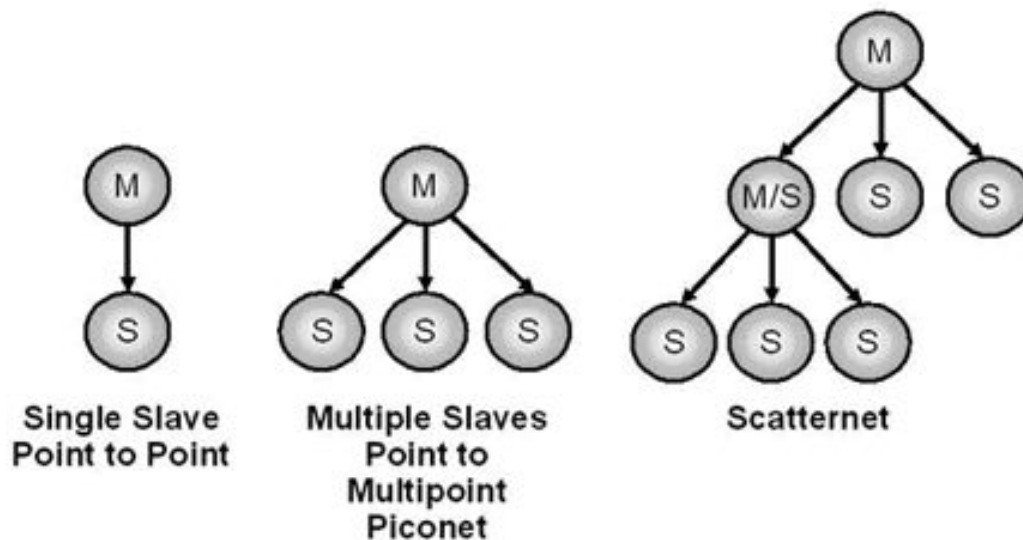


Imagen 1 Arquitectura Red Bluetooth

1.2. Pila Protocolos.

La pila del protocolo BLE se divide en tres partes básicas: Controller, Host y Applications.

El Controller es el dispositivo físico que permite transmitir y recibir señales radio e interpretarlas como paquetes con información.

El Host es la pila de software que administra cómo dos o más dispositivos se comunican entre ellos. No está definida ninguna interfaz superior para el Host.

Aplicaciones utilizan la pila de software, y a su vez ésta utiliza el controlador. detalles en los experimentos que se han realizado y que se analizarán más adelante.

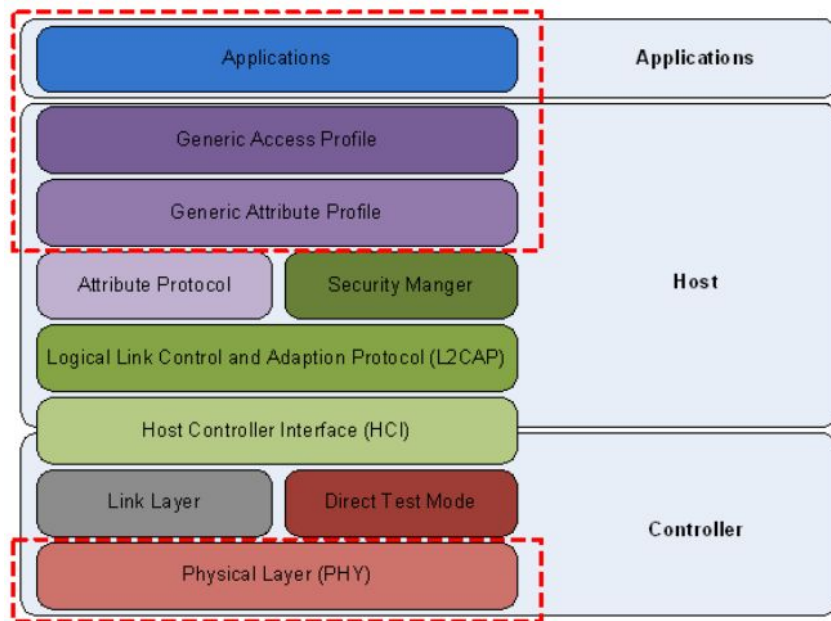


Imagen 2 Protocolo de Capas Bluetooth

Capa Física

La capa física es la que se encarga de enviar las señales al aire, transmitiendo y recibiendo bits usando ondas radio en la banda de frecuencia Industrial Scientific Medical (ISM) 2.4 Ghz que se extiende desde 2402 MHz hasta 2480 MHz. La separación entre los 40 canales utilizados es de 2 MHz (numerados de 0 a 39 y de 1 MHz de anchura cada uno). Existen 3 canales dedicados para el Advertising y 37 para la transmisión de datos. Los canales 37, 38, y 39 son usados sólo para el envío de paquetes de Advertising. El resto son usados para el intercambio de datos durante la conexión.

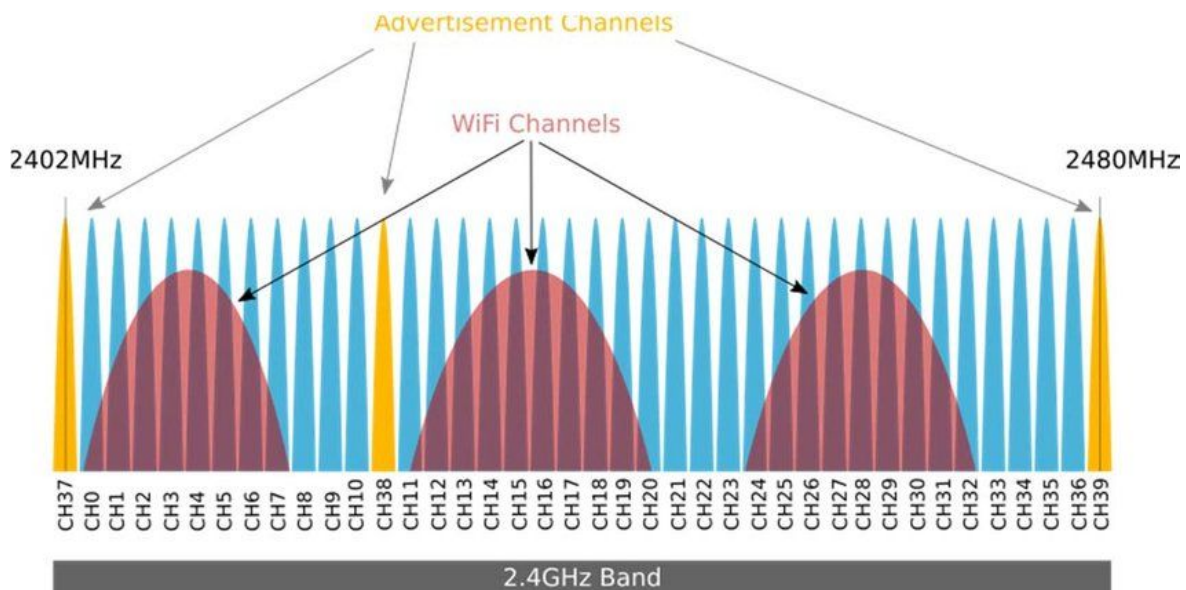


Imagen 3 Canales utilizados por Ble

Capa de Enlace

Esta es la capa responsable de los estados de Advertising, Scanning, creación y mantenimiento de las conexiones. También es la responsable de la estructura de los paquetes.

1.3. Detección de Estados

Existen cinco modos básicos en los que un dispositivo BLE puede operar.

Standby: Básicamente el dispositivo ni transmite ni recibe. Por lo general, este estado está asociado con un sistema durmiente para conservar energía.

Scanning: Se refiere a escuchar a paquetes de Advertising enviados a través de sus canales. Este modo se usa para explorar dispositivos.

Initiating: En general, este estado es el estado al que entra el dispositivo central antes de pasar a estado de conexión. El dispositivo central escucha los Advertising de periféricos,

pero una vez recibe el Advertising del periférico deseado, el central debe conectar enviando los datos correctos

Conexión: El dispositivo se encuentra conectado a otro en 37 canales reservados para la transmisión de datos. En este modo, los dispositivos conectados asumen dos roles diferentes. El dispositivo que se encontraba en modo initiating asume el rol de maestro, que es el encargado del establecimiento y la sincronización de la conexión con uno o varios dispositivos. El que se encontraba en Advertising asume el rol de esclavo y sólo se comunicará con el maestro.

Advertising: El dispositivo que tiene el rol de periférico entra en estado de donde envía paquetes en los canales de Advertising. En este estado también escucha cualquier respuesta (solicitud) de los paquetes desde el dispositivo central. Este modo es de los más críticos para analizar desde el punto de vista de la potencia ya que el dispositivo periférico tardará más o menos un tiempo de Advertising (anunciándose) dependiendo de la aplicación. Hay que tener en cuenta que el tiempo de transmisión afecta al consumo de energía, por tanto el intervalo de Advertising afecta directamente al consumo de potencia y la vida de las baterías.

El dispositivo Scanner busca en los canales de Advertising paquetes de Advertising de otros dispositivos. Existen cuatro tipos de Advertising: General, Directed, Nonconnectable y Discoverable.

1.4. Evento advertising

Connectable Undirected Advertising: es el más común. El dispositivo escáner puede recibir los anuncios o ir a hacer una conexión.

Connectable Directed Advertising : se usa cuando un dispositivo necesita conectarse de manera rápida. Este Advertising debe repetirse cada 3.75 milisegundos. Se permite estar en este estado un tiempo máximo de 1.28s.

Nonconnectable Advertising: es usado por dispositivos que quieren emitir (difundir) datos. El dispositivo no puede estar detectable (no puede recibir Scan Request) ni en conexión.

Discoverable Advertising: no se puede usar para iniciar una conexión, pero si para ser escaneado por otros dispositivos y detectar Scan Request. El dispositivo escáner puede obtener datos del Advertiser ya que éste responde a cada Scan Request detectado con un Scan Response. También se puede utilizar para emitir datos.

<i>Advertising Packet Type</i>	<i>Connectable</i>	<i>Scannable</i>	<i>Directed</i>	<i>GAP Name</i>
ADV_IND	Yes	Yes	No	Connectable Undirected Advertising
ADV_DIRECT_IND	Yes	No	Yes	Connectable Directed Advertising
ADV_NONCONN_IND	No	No	No	Non-connectable Undirected Advertising
ADV_SCAN_IND	No	Yes	No	Scannable Undirected Advertising

Tabla 1 Tipos de paquetes de Advertising y sus características

Capítulo 2

2.1. Objetivo

Realizar el análisis del Intervalo de Escaneo para la detección eficiente del tráfico de los paquetes de publicidad **Noconectable** de la tecnología Bluetooth le

2.2. Desarrollo

Esta actividad busca entregar una guía de los procedimientos del Análisis de los protocolos de detección y descubrimiento de la tecnología Bluetooth BLE, el cual se elabora un informe paso a paso que busca como objetivo los resultados cualitativos representados en una plataforma estadística.

2.3. Implementación

Para el desarrollo de la actividad se utilizaron las siguientes aplicaciones :

Como soporte se utilizó un Software de Virtualización Virtualbox en esta aplicación se implementó sistema operativo Linux Ubuntu 16.04. Este permitió la configuración del dispositivo Bluetooth sin las limitantes de acceso lo que ofrece el control total del dispositivo Bluetooth. USB Bluetooth 4.0 de fabricante genérico que ofrece doble modulación y permite la configuración de BLE.

Los datos estadísticos se obtuvieron con las herramientas de monitoreo como Wireshark y tshark permitieron comunicarse con los dispositivos Bluetooth y realizar capturas de tráfico de paquetes para medición.

Bluetooth v4.0 (2010): El SIG de Bluetooth completó la especificación del Núcleo de Bluetooth en su versión 4.0 que incluye al Bluetooth clásico, el Bluetooth de alta velocidad

y los protocolos Bluetooth de bajo consumo. El Bluetooth de baja energía (Bluetooth Low Energy o BLE) es un subconjunto de Bluetooth v4.0 con una pila de protocolos completamente nueva para desarrollar rápidamente enlaces sencillos. Como alternativa a los protocolos estándar de Bluetooth que se introdujeron en Bluetooth v1.0 a v4.0, está dirigido a aplicaciones de muy baja potencia con dispositivos alimentados con pilas botón. Los diseños de los chips permiten dos tipos de implementación, de modo dual y de modo único. El 17 de diciembre de 2010, el Bluetooth SIG adoptó la tecnología Bluetooth de bajo consumo como el rasgo distintivo de la versión 4.0 que llegó a conocerse como Wibree. Los nombres provisionales Wibree y Bluetooth ULP (Ultra Low Power) fueron abandonados y el nombre BLE comenzó a ser utilizado ampliamente. A finales de 2011, se presentaron los nuevos logotipos Smart Bluetooth Ready para los anfitriones y Smart Bluetooth para los sensores de menor tamaño basados en la tecnología BLE

2.3. Configuración

Se realizó la configuración y virtualización del sistema operativo Ubuntu versión 16.04 desktop con entorno GUI se adjunta link de descarga del Distro como también el manual de instalación de la pagina Ubuntu <https://ubuntu.com/tutorials/tutorial-install-ubuntu-desktop-1604#1-overview>

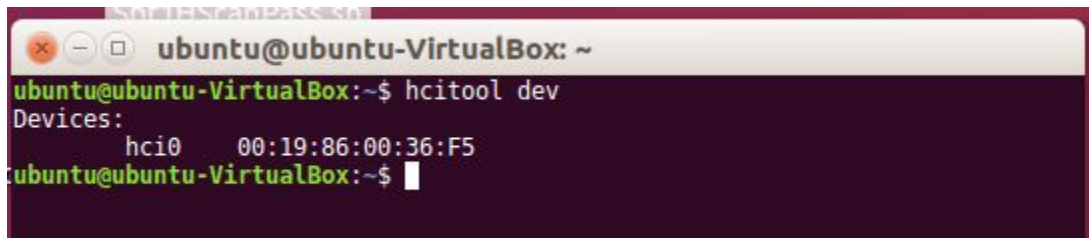
continúa con la instalación de dispositivo bluetooth instalando los controladores para acceso al medio de la interfaz física USB con la herramienta Bluez

```
ubuntu@ubuntu-VirtualBox:~$ sudo snap install bluez
[sudo] password for ubuntu:
2020-06-23T11:18:51-05:00 INFO Waiting for restart...
Setup snap "bluez" (204) security profiles

bluez 5.47-4 from 'canonical' installed
```

que proporciona soporte para la configuración en la capas y acceso a los principales protocolos de Bluetooth LE. Es flexible, eficiente y utiliza una implementación modular.

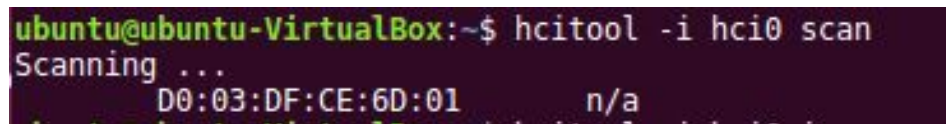
Se inició la comprobación con los parámetros del USB Bluetooth con el comando `hcitool` el cual activa dispositivo se puede identificar la BD_ADDR (Bluetooth Device Address) dirección de enlace muy similar a las MAC y su nombre de identificación.



```
ubuntu@ubuntu-VirtualBox: ~  
ubuntu@ubuntu-VirtualBox:~$ hcitool dev  
Devices:  
hci0    00:19:86:00:36:F5  
ubuntu@ubuntu-VirtualBox:~$
```

los comando ejecutados con `hcitool` permiten acceder al parámetros de Host Controller interface La capa de interfaz del controlador del host (HCI) es una capa que transporta comandos y eventos entre el host y los elementos del controlador de la pila de protocolos Bluetooth

por ejemplo para realizar un escaneo de otros dispositivos



```
ubuntu@ubuntu-VirtualBox:~$ hcitool -i hci0 scan  
Scanning ...  
D0:03:DF:CE:6D:01    n/a
```

El comando de `hcitool` más utilizado en este trabajo va a ser `cmd` que permite configurar y ejecutar todos los parámetros HCI de los dispositivos Bluetooth

El HCI Command, el primer campo del paquete es de dos bytes e indican el código del comando, estos dos bytes se dividen en dos partes:

la primera de 6 bits es para el campo de código de grupo o OGF (OpCode Group Field) la segunda de 10 bits indica el campo de código de comando o OCF (OpCode Command Field).

```

▼ Bluetooth HCI Command - LE Set Scan Enable
  ▼ Command Opcode: LE Set Scan Enable (0x200c)
    0010 00.. .... = Opcode Group Field: LE Controller Commands (0x08)
    .... ..00 0000 1100 = Opcode Command Field: LE Set Scan Enable (0x00c)
    Parameter Total Length: 2
    Scan Enable: true (0x01)
    Filter Duplicates: false (0x00)
    [Response in frame: 2]
    [Command-Response Delta: 1,61ms]

```

```

ubuntu@ubuntu-VirtualBox:~$ sudo hcitool -i hci0 cmd 0x08 0x000C 01 00
< HCI Command: ogf 0x08, ocf 0x000c, plen 2
  01 00
> HCI Event: 0x0e plen 4
  01 0C 20 01
ubuntu@ubuntu-VirtualBox:~$

```

Se actualiza los parámetros del dispositivo ejecutando los comandos con la herramienta hcitool que se instaló previamente la primera configuración deshabilita el escaneo para interfaz nativa de Bluetooth, los siguientes parámetros habilita el escaneo para Bluetooth HCI Command - LE Set Scan Enable específico de baja energía.

```
Sudo hciconfig -a hci0 noscan
```

2.4 Configuración Intervalo de escaneo

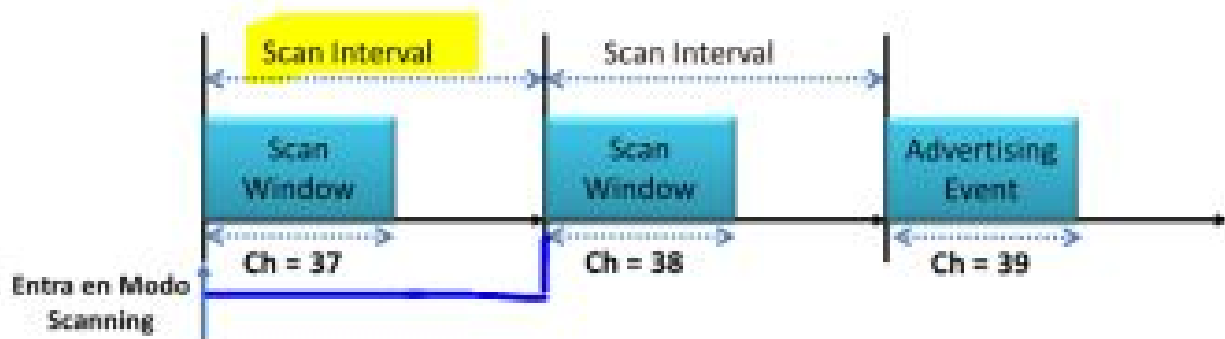
```

sudo hcitool -i hci0 cmd 0x08 0x000C 00 00
sudo hcitool -i hci0 cmd 0x08 0x000B 00 10 00 10 00 00 00
sudo hcitool -i hci0 cmd 0x08 0x000C 01 00

```

En el segunda configuración se busca igualar el intervalo del scan al intervalo de la ventana esto con la finalidad de ejecutar un escaneo continuo y evitar un estado standby que se presenta cuando deja de realizar el escaneo. sabiendo que el escaneo pasivo real se

realiza durante el escaneo de la ventana.



La diferencia entre el escaneo activo y el pasivo es que active escaneos active solicitan un SCAN_RESPONSE al anunciante. Esto se realiza enviando un SCAN_REQUEST después de que se hayan detectado anuncios.

El ultimo parametro inicia el escaneo , en el momento en el que se habilita el proceso continúa realizando la actividad hasta cuando nuevamente se desactiva con un nuevo comando.

La captura de la información se hace a partir de la herramienta de wireshark.

Una vez configurado los parámetros del escaneo con los intervalos que para las muestra se habilita el sniffer que captura la información de los eventos que procesen al inicio del escaneo.

```
▶ Frame 3: 11 bytes on wire (88 bits), 11 bytes captured (88 bits) on interface 0
▶ Bluetooth
▶ Bluetooth HCI H4
▼ Bluetooth HCI Command - LE Set Scan Parameters
  ▶ Command Opcode: LE Set Scan Parameters (0x200b)
    Parameter Total Length: 7
    Scan Type: Passive (0x00)
    Scan Interval: 80 (50 msec)
    Scan Window: 16 (10 msec)
    Own Address Type: Public Device Address (0x00)
    Scan Filter Policy: Accept all advertisements. Ignore directed advertisements not addressed to this device (0x00)
    [Response in frame: 4]
    [Command-Response Delta: 1,152ms]
```


El sesgo de las muestras se realiza con en la ejecución de los comando que habilita y deshabilitando el escaneo en la recolección de los paquetes que se pretenden obtener en esta caso se realizaron muestra de 1000 paquetes.

```

ubuntu@ubuntu-VirtualBox: ~
01 0C 20 00
ubuntu@ubuntu-VirtualBox:~$ sudo hcitool -i hci0 cmd 0x08 0x000C 00 00
< HCI Command: ogf 0x08, ocf 0x000c, plen 2
00 00
> HCI Event: 0x0e plen 4
01 0C 20 00
ubuntu@ubuntu-VirtualBox:~$ sudo hcitool -i hci0 cmd 0x08 0x000C 01 00
< HCI Command: ogf 0x08, ocf 0x000c, plen 2
01 00
> HCI Event: 0x0e plen 4
01 0C 20 00
ubuntu@ubuntu-VirtualBox:~$ sudo hcitool -i hci0 cmd 0x08 0x000C 00 00
< HCI Command: ogf 0x08, ocf 0x000c, plen 2
00 00
> HCI Event: 0x0e plen 4
01 0C 20 00
ubuntu@ubuntu-VirtualBox:~$ sudo hcitool -i hci0 cmd 0x08 0x000B 00 16 00 1
0 00 00 00
< HCI Command: ogf 0x08, ocf 0x000b, plen 7
00 16 00 10 00 00 00
> HCI Event: 0x0e plen 4
01 0B 20 00
ubuntu@ubuntu-VirtualBox:~$ sudo hcitool -i hci0 cmd 0x08 0x000B 00 50 00 1
0 00 00 00

```

la aplicación va recolectando los paquetes como se observar en el modo gráfico de la herramienta wireshark , el escaneo y se aplica los filtro únicamente para la visualización del dispositivo.

Cap1000pakscanonavdInt250.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
6	0.244946	controller	host	HCI_EVT	43	Rcvd LE Meta (LE Advertising Report)
7	0.003949	controller	host	HCI_EVT	43	Rcvd LE Meta (LE Advertising Report)
8	0.039880	controller	host	HCI_EVT	46	Rcvd LE Meta (LE Advertising Report)
9	0.113220	controller	host	HCI_EVT	43	Rcvd LE Meta (LE Advertising Report)
10	0.003953	controller	host	HCI_EVT	43	Rcvd LE Meta (LE Advertising Report)
11	0.093053	controller	host	HCI_EVT	32	Rcvd LE Meta (LE Advertising Report)
12	0.072868	controller	host	HCI_EVT	46	Rcvd LE Meta (LE Advertising Report)
13	0.089006	controller	host	HCI_EVT	46	Rcvd LE Meta (LE Advertising Report)
14	0.043893	controller	host	HCI_EVT	43	Rcvd LE Meta (LE Advertising Report)
15	0.064020	controller	host	HCI_EVT	32	Rcvd LE Meta (LE Advertising Report)
16	0.151268	controller	host	HCI_EVT	46	Rcvd LE Meta (LE Advertising Report)
17	0.121940	controller	host	HCI_EVT	32	Rcvd LE Meta (LE Advertising Report)
18	0.160017	controller	host	HCI_EVT	39	Rcvd LE Meta (LE Advertising Report)
19	0.109312	controller	host	HCI_EVT	30	Rcvd LE Meta (LE Advertising Report)
20	0.013475	controller	host	HCI_EVT	43	Rcvd LE Meta (LE Advertising Report)
21	0.004023	controller	host	HCI_EVT	43	Rcvd LE Meta (LE Advertising Report)

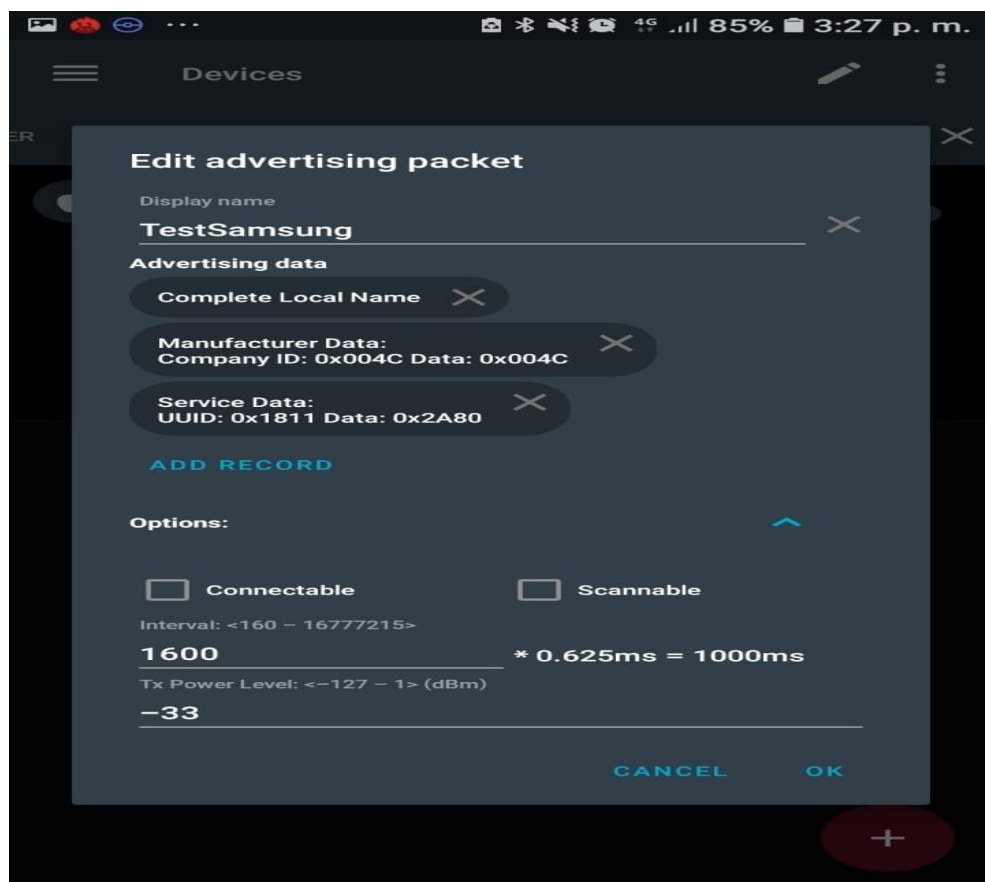
2.5. Configuración de Paquetes Advertising

Enfocando únicamente en la información de la captura del tráfico de los paquete de anunciación o Advertising de evento Non-connectable.

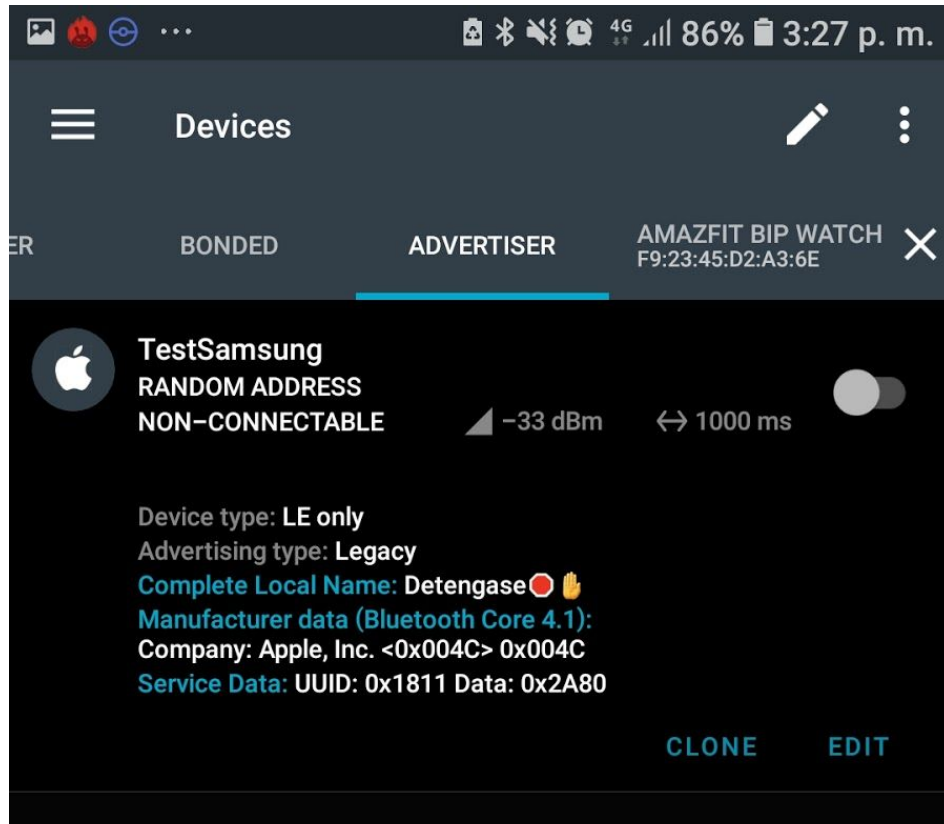
ADV_NONCONN_IND	No	No	No	Non-connectable Undirected Advertising
-----------------	----	----	----	---

En este estado los dispositivos que se encuentran en el estado de escanear reciben la información de la baliza que emite el enunciamiento o Advertising sin obtener respuesta al comunicado o realizar una conexión previa.

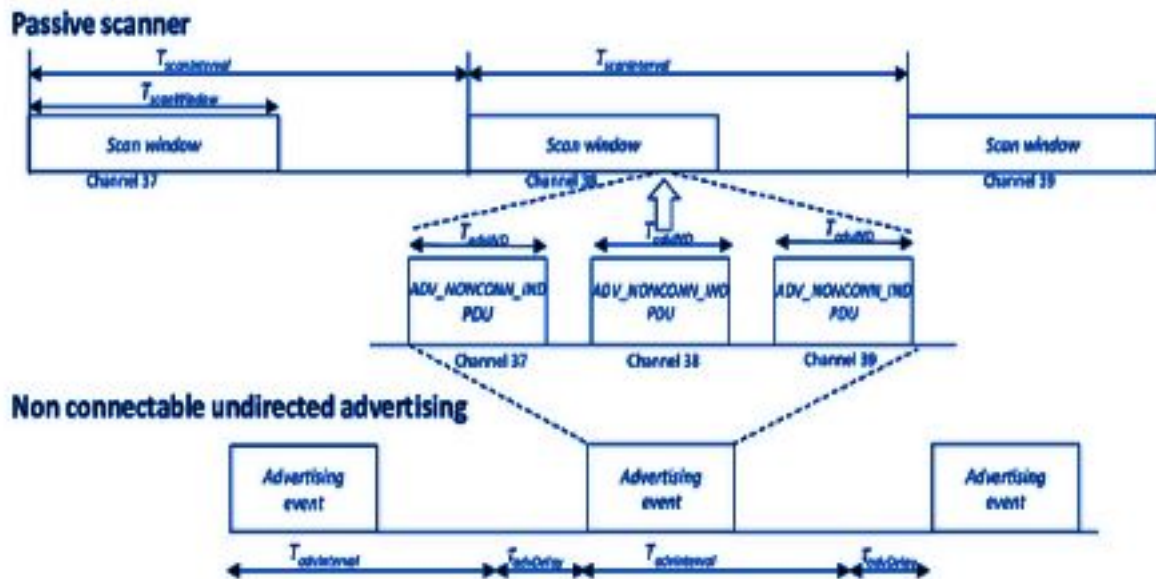
La configuración del paquete Advertising se dio mediante una herramienta móvil nRF Connect



la herramienta permite realizar la configuración del dispositivo en el modo de Advertiser donde se puede agregar los parámetros de información del anuncio en unas pestañas predeterminadas adicional existen parámetros que permite modificar los valores señal y el tiempo de del envío



Los parámetros que se establecieron para obtener las muestras varían también del envío de los datos de Advertising estos parámetros se realizaron en la herramienta móvil , como mínimo tiene que ser de uno 100 ms. El retardo es un valor aleatorio entre 0 ms y 10 ms que se añade automáticamente. Este último valor ayuda a reducir colisiones entre Advertising de dispositivos diferentes .



La información de los datos es capturada por el sniffer y con se muestra en la pantalla, se emite el anuncio tal con se realizo en la configuracion del movil

```

Bluetooth HCI Event - LE Meta
Event Code: LE Meta (0x3e)
Parameter Total Length: 43
Sub Event: LE Advertising Report (0x02)
Num Reports: 1
Event Type: Non-Connectable Undirected Advertising (0x03)
Peer Address Type: Random Device Address (0x01)
BD_ADDR: Arbiter_11:ad:e0 (65:73:e2:11:ad:e0)
Data Length: 31
Advertising Data
  Device Name: Detengase
    Length: 18
    Type: Device Name (0x09)
    Device Name: Detengase
  Manufacturer Specific
    Length: 5
    Type: Manufacturer Specific (0xff)
    Company ID: Apple, Inc. (0x004c)
    Data: 004c
      [Expert Info (Note/Undecoded): Undecoded]
  Service Data - 16 bit UUID
    Length: 5
    Type: Service Data - 16 bit UUID (0x16)
    UUID 16: Alert Notification Service (0x1811)
    Service Data: 2a80
RSSI: 0dBm

```

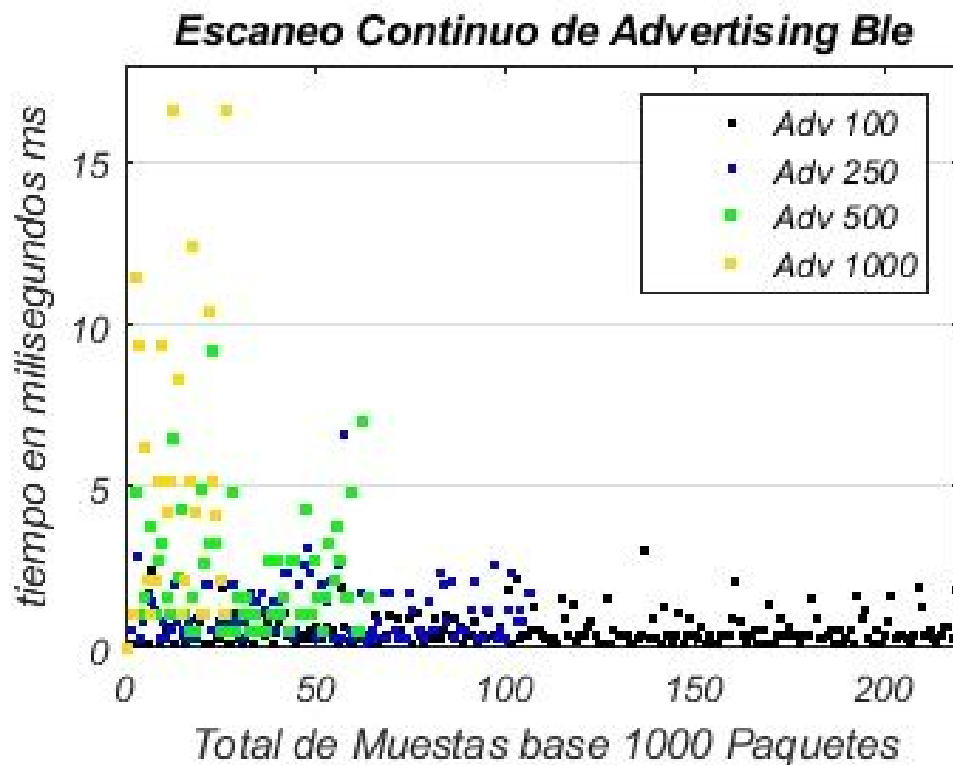
Capítulo 3

3.1. Discusión

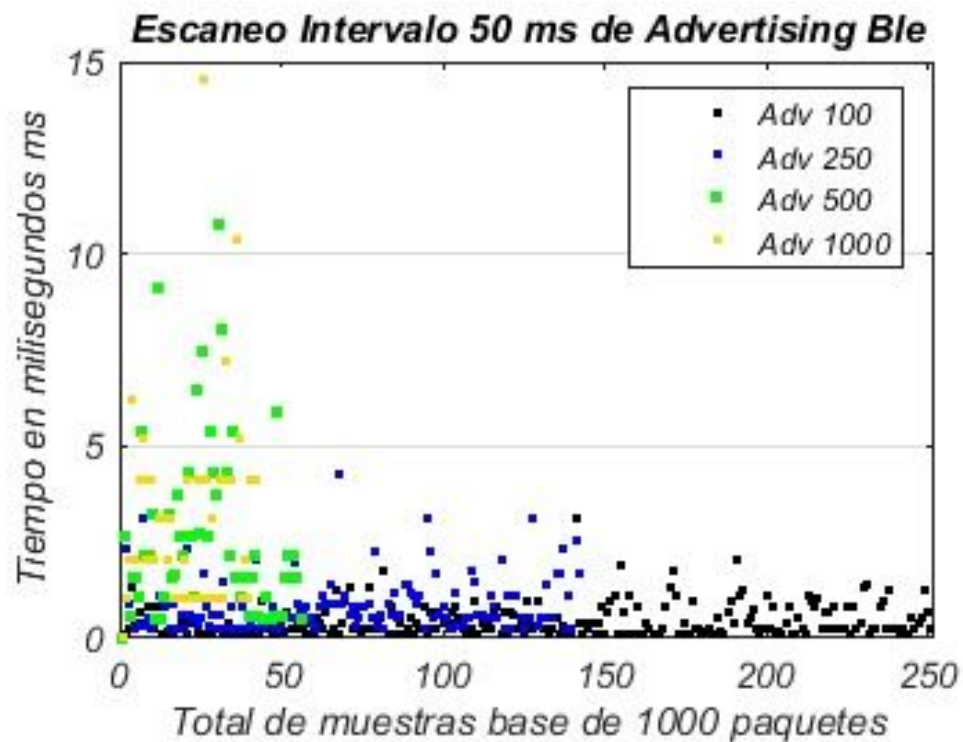
Con el objetivo del ejercicio los datos que se obtuvieron se muestran en las siguientes gráficas

Primera prueba se realiza con el escaneo continuo desde una Laptop con dispositivo USB genérico igualando la ventana de tiempo con el intervalo del escaneo a un tiempo de 10 ms para la captura de datos transmitidos por un Celular Samsung A5 que genera paquetes adv_noN-conectables con intervalos de 100, 250, 500 1000 ms donde se registran en la herramienta de escaneo entre 1000 paquetes por cada intervalo del estado de eventos.

3.2. Resultados



Como se observa en la primera gráfica los datos recolectados muestra en la ventana de tiempo el tráfico que puede obtener con la variación de los parámetros del hci entre más pequeño el intervalo mayor el Flujo y cantidad de eventos generados pero con el escaneo continuo no se logra la captura de la totalidad los paquetes como si se puede evidenciar de en los resultado con el escaneo con intervalos que aumenta la cantidad de eventos capturados disminuyendo el tiempo máximo entre capturas. .



3.3. Conclusiones

Como se presenta el ejercicio dispone de encontrar la mejor disposición para la captura de los Eventos de publicidad que emiten los dispositivos bluetooth con low energy en este caso se puede evidencia que la captura más eficiente no depende de un escaneo continuo si no de los espacio en relación de los paquetes ya estos tiempos disminuyen la colisión de paquetes . Así que se la eficiencia aumenta al considerar los tiempo de stand by entre los paquetes.

Lista de referencias

Basa en el análisis en tesis de grado :

TÍTULO DEL TFG: Bluetooth 4.0 Low Energy: Análisis de las prestaciones y aplicaciones para la automoción

AUTOR: Yassir Akhayad

FECHA: 08/02/2016

TÍTULO Covered Core Package version: 4.0

AUTOR Current Master TOC

FECHA Publication date: 30 June 2010

Web referencia

<https://docs.ubuntu.com/core/en/stacks/bluetooth/bluez/docs/install-bluez>

<https://californiaconsultants.org/wp-content/uploads/2014/05/CNSV-1205-Decuir.pdf>

<https://www.bluetooth.com/blog/bluetooth-low-energy-it-starts-with-advertising/>