ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES Y DE TELECOMUNICACIÓN

UNIVERSIDAD DE CANTABRIA

Trabajo Fin de Grado

Evaluación del rendimiento del *tethering* Bluetooth sobre máquinas virtuales

(Performance evaluation of Bluetooth *tethering* on virtual machines)

Para acceder al Título de

Graduado en Ingeniería de Tecnologías de Telecomunicación

Autor: Alberto Martín Palacios

Junio – 2018

E.T.S. DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN

GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN

CALIFICACIÓN DEL TRABAJO FIN DE GRADO

Realizado por: Alberto Martín Palacios

Director del TFG: Marta García Arranz

Título: "Evaluación del rendimiento del *tethering* Bluetooth sobre máquinas virtuales"

Title: "Performance evaluation of Bluetooth *tethering* on virtual machines"

Presentado a examen el día: 20/06/2018

Para acceder al Título de: GRADUADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN

Composición del Tribunal:

Presidente (Apellidos, Nombre): Basterrechea Verdeja, José

Secretario (Apellidos, Nombre): García Arranz, Marta

Vocal (Apellidos, Nombre): Sanz Gil, Roberto

Este Tribunal ha resuelto otorgar la calificación de: .....

Fdo.: El Presidente

Fdo.: El Secretario

Fdo.: El Vocal

TFG (sólo si es distinto del secretario)

Fdo.: El Director del

Vº Bº del Subdirector

Trabajo Fin de Grado Nº (a asignar por Secretaría)

Agradecimientos

En primer lugar, me gustaría agradecer a Marta García su disposición y amabilidad a lo largo de estos meses y el hecho de que dirija el presente Trabajo de Fin de Grado, el cual no habría sido posible sin sus sugerencias e indicaciones.

Agradecer toda la ayuda de mi compañero Alberto Fernández durante el desarrollo del proyecto en el laboratorio, al igual que a mis compañeros del Grado como Raúl, Fernando y Jorge por su apoyo.

Por último, quiero resaltar el apoyo tanto de mi familia, como amigos a lo largo de este proyecto.

Alberto Martín Palacios, Junio 2018.

Resumen

El trabajo tiene como principal objetivo la instalación y configuración de una máquina virtual Linux con todos aquellos paquetes y herramientas necesarios para el estudio del perfil PAN (Personal Area Network) de Bluetooth. Dicho perfil hace uso del protocolo BNEP (Bluetooth Network Encapsulation Protocol), que permite el transporte del tráfico IP sobre dicha interfaz inalámbrica. En particular, se trata de caracterizar el rendimiento del escenario basado en un punto de acceso a red o NAP (Network Access Point) que permite la conexión desde el dispositivo Bluetooth a otras redes, como es el caso de Internet. Este procedimiento es también conocido como tethering o “anclaje a red”. La decisión de llevar a cabo este trabajo utilizando máquinas virtuales se debe a que se prevé su utilización en las prácticas de la asignatura “Redes no convencionales” de 4º curso del Grado en Ingeniería de Tecnologías de Telecomunicación, mención de telemática.

Abstract

The main objective of the work is the installation and the Bluetooth software configuration with all the packages and tools necessary for the Bluetooth PAN (Personal Area Network) profile. This profile makes use of the BNEP protocol (Bluetooth Network Encapsulation Protocol), which allows the transport of IP traffic over said wireless interface. In particular, it is a scenario based on a network access point or NAP (network access point) that allows connection from the Bluetooth device to other networks, such as the Internet. This is also known as tethering or "network anchoring". The decision to carry out this work on virtual machines is due to the fact that they will be used in the practical lessons for the subject “Non-conventional networks” of the 4th year of the Degree in Telecommunications Technology Engineering, mention of telematics.

Índice general

Índice de figuras

Índice de tablas

Acrónimos

ACL: Asynchronous Connection-oriented [logical transport]

AMP: Alternate MAC/PHY

A2DP: Advanced Audio Distribution Profile

BD_ADDR: Bluetooth Device Address

BNEP: Bluetooth Network Encapsulation Protocol

BR: Basic Rate

EDR: Enhanced Data Rate

FHSS: Frequency Hopping Spread Spectrum

FTP: File Transfer Profile

GAP: Generic Access Profile

GATT: Generic Attribute Profile

GFSK: Gaussian Frequency Shift Keying

GOEP: Generic Object Exchange Profile

HCI: Host Controller Interface

HFP: Hands-Free Profile

HSP: HeadSet Profile

IPv4: Internet Protocol version 4

IPv6: Internet Protocol version 6

L2CAP: Logical Link Control and Adaptation Protocol

LC: Link Controller

LE: Low Energy

LM: Link Manager

LMP: Link Manager Protocol

MTU: Maximum Transmission Unit

NAP: Network Access Point

NAT: Network Address Translation

OBEX: OBject EXchange protocol

PAN: Personal Area Network

PANU: Personal Area Network User

PSM: Protocol Service Multiplexer

RF: Radio Frequency

RFC: Request For Comments

RFCOMM: Serial Cable Emulation Protocol

SDP: Service Discovery Protocol

SPP: Serial Port Profile

TCP: Transmision Control Protocol

TDD: Time-Division Duplex

UDP: User Datagram Protocol

UUID: Universal Unique Identifier

VDP: Video Distribution Profile

Palabras clave

Bluetooth, SIG, BNEP, Perfil PAN, NAP, PANU, Rendimiento, WPAN, Throughput, NTTCP, IPERF, BLUEZ, Dongle USB, Wireshark, Bluetoothctl, Blueman.

Capítulo 1. Introducción

En la actualidad, nos encontramos en un mundo en el que la necesidad de comunicarse es algo fundamental y las comunicaciones deben ser posibles desde cualquier lugar, por ello no deben estar limitadas a soluciones de conexión cableadas, de ahí el auge de las conexiones inalámbricas en las últimas décadas.

Bluetooth surge de la necesidad de empresas de informática y de telecomunicaciones de desarrollar una interfaz abierta y de bajo coste para facilitar la comunicación entre dispositivos sin la utilización de cables, aprovechando la movilidad de los dispositivos inalámbricos. [1]

Es un enlace de radiofrecuencia de corto alcance que permite la transmisión de voz y datos debido a su sencillez entre varios dispositivos (ordenador, teléfonos móviles, pulseras…), que forman una pequeña red inalámbrica llamada red de área personal inalámbrica (WPAN).

Los principales objetivos de Bluetooth son principalmente estos tres:
- Facilitar las comunicaciones entre dispositivos móviles.
- Eliminar los cables y los conectores de los dispositivos.
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre equipos personales.

Bluetooth nace al principio de 1998, de la colaboración de varias empresas líderes de la industria de las TIC: Ericsson, Nokia, Intel, IBM, Toshiba, Motorola y, más tarde, 3Com (Palm), que constituyeron el SIG (Special Interest Group), al que actualmente ya pertenecen más de 9000 compañías de telecomunicaciones.

El desarrollo de la tecnología inalámbrica Bluetooth es dirigido por los miembros del SIG, además de implementar y comercializar la tecnología en sus productos aunque el SIG por sí mismo no fabrica ni vende dispositivos Bluetooth. Por ello, la evolución de Bluetooth ha sido tan constante durante el siglo XXI, aumentando las tasas de transferencia de datos, el rango de alcance, mejorando la gestión de la energía y, por supuesto, aumentando la compatibilidad entre dispositivos. Existen varias versiones desde la 1.0 que fue lanzada en el año 2002 hasta la más actual que es la 5.0, activa desde finales del año 2016. En la tabla siguiente se muestran los aspectos más relevantes de las distintas especificaciones de Bluetooth.

Especificaciones	Año	Características
Bluetooth 1.0	2002	Primera versión usada para la transmisión de datos con una tasa aproximada de 721 Kbit/s. Enfrentó muchos problemas de comunicación entre dispositivos.
Bluetooth 1.2	2003	Realizan una conexión más rápida y pueden detectar otros dispositivos con Bluetooth (Discovery)
Bluetooth 2.0 + EDR	2004	Introducción de una tasa de datos mejorada (EDR) opcional para acelerar la transferencia de datos (x3 de velocidad)
Bluetooth 2.1 + EDR	2007	Incluye Secure Simple Pairing (SSP) que mejora la experiencia de emparejamiento de dispositivos Bluetooth, mientras que aumenta el uso y la fuerza de seguridad.
Bluetooth 3.0 HS	2009	Soporta velocidades teóricas de transferencia de datos de hasta 24 Mbit/s entre sí, aunque no a través del enlace Bluetooth propiamente dicho. La conexión Bluetooth nativa se utiliza para la negociación y el establecimiento mientras que el tráfico de datos de alta velocidad se realiza mediante un enlace 802.11

Bluetooth 4.0	2010	Surge el Bluetooth de baja energía (Bluetooth Low Energy o BLE) con una pila de protocolo completamente nueva para desarrollar rápidamente enlaces sencillos.
Bluetooth 4.1	2013	El SIG inicia el camino hacia la IoT (Internet of Things)
Bluetooth 4.2	2014	Se introducen mejoras en conectividad IP y en seguridad. El 90% de los móviles son Bluetooth.
Bluetooth 5.0	2016	Cobertura x4, velocidad x2 y capacidad de difusión x8. Se añade la capacidad de Mesh Networking que permite el despliegue de redes con muchos dispositivos BLE.

Tabla 1. Características de las distintas especificaciones Bluetooth

En la actualidad los Smartphones se están convirtiendo rápidamente en la principal plataforma de comunicación informática y de datos. Están equipados con Bluetooth y WIFI, que complementan sus capacidades de comunicación celular. Bluetooth se colocó originalmente en teléfonos móviles para la comunicación de área personal, como auriculares inalámbricos, sincronización con un PC cercano y para el anclaje a la red o *tethering* (proceso por el cual un dispositivo móvil con conexión a Internet actúa como pasarela para ofrecer acceso a la red a otros dispositivos). [2]

Las comunicaciones locales pueden realizarse potencialmente a través de WIFI o mediante Bluetooth. Al tener estas alternativas, las consideraciones importantes son el rendimiento y el consumo de energía que se pueden obtener en cada uno, debido a que la comunicación inalámbrica es uno de los principales causantes del consumo de batería en un teléfono móvil.

Como Bluetooth se planificó para la comunicación de área personal, su alcance de transmisión es mucho más corto que WIFI. Por lo tanto, comparar los dos es significativo solo cuando los dispositivos están lo suficientemente cerca para que ambos puedan ser utilizados. Dado que Bluetooth también ofrece un ancho de banda menor, uno podría preguntarse por qué molestarse con él en absoluto. La razón para seguir considerando Bluetooth aún es que los estudios previos de potencia han sugerido que Bluetooth es mucho más eficiente que WIFI, lo que motivó múltiples trabajos en combinaciones inteligentes de Bluetooth y WIFI.

Existen estudios del rendimiento de una WPAN Bluetooth, en términos del retardo extremo a extremo y del throughput, contemplando el uso de perfiles y evaluando el efecto de la configuración de la calidad de servicio y de las retransmisiones. [3]

El estudio desarrollado en este proyecto se basará en la evaluación del rendimiento. El modelo propuesto ha sido el del perfil PAN basado en BNEP (Bluetooth Network Encapsulation Protocol), contemplando el overhead introducido por los niveles superiores. Existen otros estudios para conectarse a una LAN con Bluetooth que se basan en el perfil SPP (Serial Port Profile), usando RFCOMM (Radio Frecuency Communication) pero que en la actualidad han quedado en un segundo plano y no se entrará en ellos

1.1. Motivación y objetivos

La motivación para enfocarme en este proyecto es la de utilizar la tecnología Bluetooth para implementar un escenario del perfil PAN (Personal Area Network), concretamente el basado en un punto de acceso a Internet, en el laboratorio docente de telemática en la ETSIIT y que se pueda usar en las prácticas de la asignatura de 4º curso, Redes no convencionales del Grado en Ingeniería de Tecnologías de Telecomunicación, analizando asimismo el rendimiento del tráfico IP sobre dicho escenario.

Por ello se realiza sobre máquinas virtuales, para que la configuración sea estable independientemente de las actualizaciones propias de un sistema operativo y de que haya que tener los permisos de administrador.

1.2. Estructura del documento

- **Capítulo 1:** Introducción

Se realiza una breve introducción sobre la tecnología Bluetooth y se explican los objetivos que se persiguen en la realización de este proyecto.

- **Capítulo 2:** Descripción general de la tecnología Bluetooth

Se realiza una explicación bastante general de la tecnología Bluetooth, como funciona, sus modos de operación, etc…

- **Capítulo 3:** Protocolo de encapsulación de red Bluetooth (BNEP)

Se realiza una explicación teórica del protocolo BNEP, que es el protocolo sobre el que va IP. Además se analizará la encapsulación de paquetes y el formato de estos.

- **Capítulo 4:** Perfil PAN

Se realiza una explicación teórica sobre el perfil PAN que es el que describe cómo debe usarse BNEP para proporcionar capacidades de red a los dispositivos Bluetooth.

- **Capítulo 5:** Generación de la máquina virtual

En este capítulo comienza la parte práctica del trabajo y se explica lo que se tiene que instalar para preparar el entorno sobre el que se trabajará.

- **Capítulo 6:** Despliegue de la red

Se realiza una explicación de la red que se va a formar, la configuración de los dispositivos para que éstos formen la red, el establecimiento de la conexión de los dispositivos y las medidas de rendimiento que se tomaron pertinentes.

- **Capítulo 7:** Conclusiones y líneas futuras de trabajo

En este capítulo se analizan las conclusiones que se han sacado realizando este trabajo y de lo que este trabajo puede servir en un futuro.

Capítulo 2. Descripción general de la tecnología Bluetooth

La tecnología inalámbrica Bluetooth es un sistema de comunicaciones de corto alcance, destinado a reemplazar los cables que conectan los dispositivos. Las características fundamentales de esta tecnología son la robustez, el bajo consumo de energía y el bajo coste.

En la actualidad existen dos especificaciones principales en la tecnología Bluetooth:

- La especificación BR (Velocidad básica) incluye "optional Enhanced Data Rate" (EDR) junto con la especificación, "Alternate Media Access Control" (AMP) y "Physical (PHY) layer extensions". Ofrece conexiones síncronas y asíncronas con velocidades brutas de 1 Mb/s para BR, 3 Mb/s para velocidad de datos mejorada (EDR) y la operación de alta velocidad hasta 54 Mb/s con el 802.11 AMP.

- La especificación LE (Low Energy) que incluye características diseñadas para habilitar productos que requieren un menor consumo de corriente, menor complejidad y costo más bajo que BR / EDR. El sistema LE también está diseñado para casos de uso y aplicaciones con menores velocidades de datos y tiene ciclos de trabajo más bajos.

Como resumen, la figura siguiente muestra las diferencias fundamentales entre la especificación BR/EDR y la especificación LE. [4]

Characteristics	Bluetooth BR/EDR	Bluetooth LE
RF Physical Channels	79 channels, 1 MHz channel spacing	40 channels, 2 MHz channel spacing
Discovery/Connect	Inquiry/Paging	Advertising
Number of Piconet slaves	7 active and 255 inactive devices Supports Scatternet	Unlimited, Does not support Scatternet
Device Address Privacy	None	Private Device Addressing Available
Max Physical Data Rate	1-3 Mbps	1 Mbps via GFSK modulation
Max Throughput	0.7-2.1 Mbps	0.27 Mbps
Encryption Algorithm	E0/SAFER+	AES-CCM
Typical Range	30 m	50m
Max Output Power	100 mW (20 dBm)	10 mW(10dBm)

Figura 1. Diferencias fundamentales entre el sistema BR / EDR y el sistema LE

2.1. Pila de protocolos Bluetooth

Como cualquier otra tecnología de comunicaciones, Bluetooth presenta una arquitectura en capas. En un nivel alto, la arquitectura se puede representar como se muestra en la figura 2, que consta de capas inferiores y superiores, perfiles y aplicaciones. Las capas inferiores o el controlador son responsables de las operaciones de bajo nivel, como descubrir dispositivos en las proximidades, realizar conexiones, intercambiar paquetes de datos, gestionar la seguridad y la energía entre otras. La funcionalidad de las capas inferiores generalmente se implementa en un chip Bluetooth. [5]

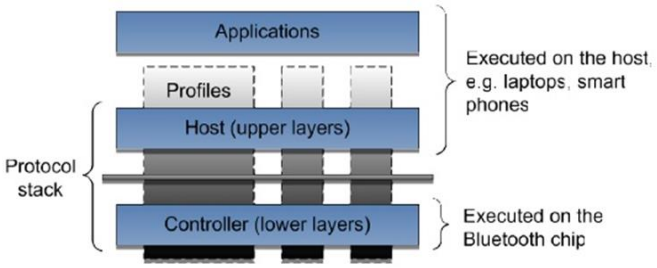

Figura 2. Representación de alto nivel de la arquitectura Bluetooth

Las capas superiores hacen uso de la funcionalidad proporcionada por las capas inferiores para proporcionar una funcionalidad más compleja. Algunos ejemplos son la emulación de puertos serie o la transferencia de grandes fragmentos de datos por perfiles que se pueden considerar como divisiones verticales a través de la pila de protocolos.

Las aplicaciones Bluetooth son las interfaces de usuario que permiten al usuario final hacer uso de la funcionalidad Bluetooth. Ejemplos de tales aplicaciones son: la exploración y transferencia de archivos, la transmisión de audio / voz, la búsqueda de otros dispositivos Bluetooth en los alrededores, etc. Las capas superiores y las aplicaciones se implementan y ejecutan en el host (portátiles y teléfonos inteligentes).

La tecnología Bluetooth se diseñó con el objetivo de reutilizar la mayor cantidad posible de estándares disponibles en lugar de diseñar todo desde cero. Algunos componentes útiles de los estándares existentes se adaptaron según fue necesario y se reutilizaron. Los componentes restantes de la pila de protocolos se diseñaron desde cero. Los componentes patentados de Bluetooth se denominan protocolos centrales, mientras que los componentes reutilizados se denominan protocolos adoptados.

La figura 3 muestra una representación detallada de la arquitectura Bluetooth. Se observa que la arquitectura Bluetooth está hecha de una mezcla de protocolos centrales y adoptados. Además, la figura también muestra un subconjunto de perfiles definidos. Hay que tener presente que aunque esta figura contiene más detalles en comparación con la figura 2, los componentes mostrados representan solo un subconjunto de todos los componentes de los que está hecha la arquitectura Bluetooth. Está más allá del alcance de esta introducción cubrir la tecnología Bluetooth completa, sino brindar una idea inicial de la misma.

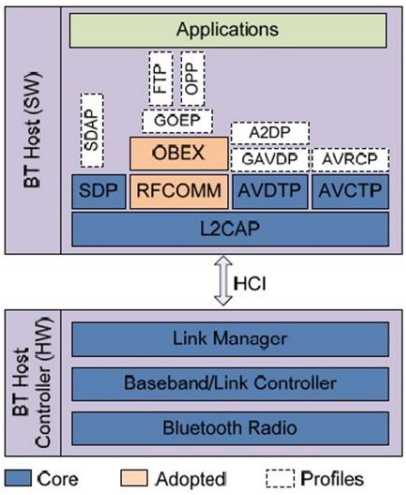

Figura 3. Representación detallada de la arquitectura Bluetooth

Las siguientes secciones están dedicadas a explicar algunos de los componentes en la arquitectura con más detalle.

2.2. Radio Bluetooth

La capa RF en Bluetooth es responsable de la transmisión y recepción de paquetes a través de la frecuencia de radio. Opera en la banda ISM de 2,4 GHz utilizando 1 MHz de ancho de banda. Es la misma banda que muchos otros dispositivos electrónicos de usuario, incluidos WIFI, microondas, teléfonos inalámbricos. Para mitigar el riesgo de interferencia, la radio Bluetooth utiliza la técnica de espectro ensanchado por salto de frecuencia (FHSS). En lugar de utilizar una frecuencia constante para enviar y recibir datos, los dispositivos de comunicación utilizan un conjunto de frecuencias y saltan rápidamente de una frecuencia a otra utilizando un patrón pseudoaleatorio. [5]

Para soportar comunicaciones bidireccionales (transmisión dúplex completa), los canales Bluetooth usan duplexación por división de tiempo (TDD) que es una técnica para convertir un canal simplex en un canal dúplex separando las señales enviadas y recibidas en intervalos de tiempos diferentes sobre el mismo canal usando acceso múltiple por división de tiempo. El uso tanto de FHSS como de TDD implica que los paquetes se transmiten en intervalos de tiempo definidos (Time Division) y en frecuencias definidas.

En el interfaz radio en Bluetooth se pueden soportar dos tipos de modulación:

- Velocidad básica (BR): este modo es obligatorio y debe ser compatible con todas las versiones de Bluetooth. BR utiliza la modulación digital GFSK (Gaussian Frecuency Shift Keying) que proporciona una tasa bruta de datos de 1 Mbps.

- Velocidad de datos mejorada (EDR): este modo es opcional y utiliza la modulación π/4-DQPSK para trabajar a 2 Mb/s y la 8-DPSK para alcanzar una velocidad bruta de 3 Mb/s.

2.3. Banda base

El controlador de banda base es responsable de las siguientes funciones principales: [5]

- Gestión de canales físicos y enlaces
- Selección de la próxima frecuencia de salto para transmitir y recibir paquetes
- Formación de piconet y scatternet
- Creación de paquetes
- Escaneo de consulta e investigación
- Conexión y escaneo de página
- Seguridad, incluido el cifrado
- Administración de energía

2.3.1. Descripción general

El sistema Bluetooth proporciona una conexión punto a punto o una conexión punto a multipunto como vemos en los apartados a) y b) de la figura 4. En una conexión punto a punto, el canal físico se comparte entre dos dispositivos Bluetooth. En una conexión punto a multipunto, el canal físico se comparte entre varios dispositivos Bluetooth. Dos o más

dispositivos que comparten el mismo canal físico forman una piconet. Un dispositivo Bluetooth actúa como el maestro de la piconet, mientras que el otro dispositivo actúa como esclavo (s). Hasta siete esclavos pueden estar activos en la piconet. El acceso al canal está controlado por el maestro. Un número ilimitado de esclavos puede recibir datos en el canal físico.

Los piconets que tienen dispositivos comunes se llaman scatternet. Cada piconet solo tiene un único maestro, sin embargo, los esclavos pueden participar en diferentes piconets sobre una base multiplexada por división de tiempo. Además, un maestro en una piconet puede ser un esclavo en otras piconets. Los piconets no se sincronizarán en frecuencia y cada piconet tiene su propia secuencia de salto.

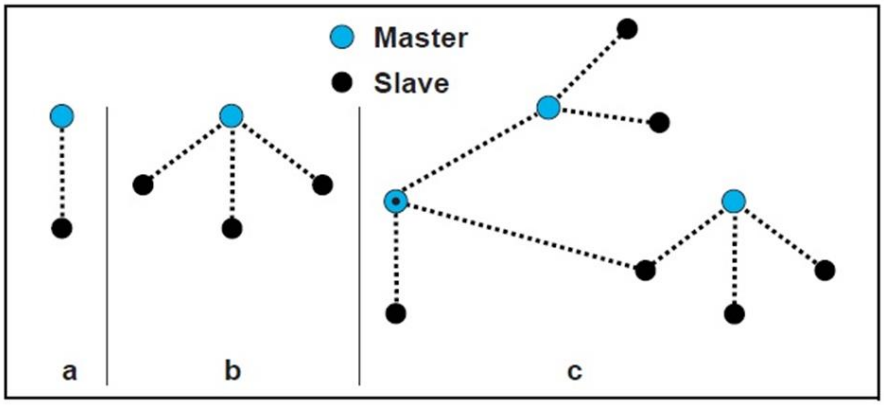

Figura 4. Piconet con un esclavo único, piconet con múltiples esclavos y scatternet

2.3.2. Canal físico

Los canales físicos se definen por una secuencia básica de salto de canal de radiofrecuencia pseudoaleatorio, el tiempo del paquete (ranura) y un código de acceso. La fase en la secuencia de salto está determinada por el reloj Bluetooth.

Todos los canales físicos se subdividen en intervalos de tiempo cuya longitud es diferente dependiendo del canal físico. Dentro del canal físico, cada evento de recepción o transmisión está asociado con un intervalo de tiempo o intervalos de tiempo. La velocidad máxima de salto es 1600 saltos / s en el estado de conexión, en el subestado del tren de sincronización y en el subestado de exploración de sincronización y el máximo 3200 saltos / s en los subestados de consulta y página.

Se definen los siguientes canales físicos:

- canal físico básico de piconet
- canal físico adaptado de piconet
- canal físico de escaneo de página
- canal físico de exploración de consulta
- canal físico de escaneo de sincronización

El canal físico básico de piconet se caracteriza por un salto pseudoaleatorio a través de los 79 canales de RF. El salto de frecuencia en el canal físico de piconet viene determinado por el reloj Bluetooth y la dirección BD_ADDR del maestro. Cuando se establece la piconet, el reloj maestro se comunica con los esclavos. Cada esclavo agregará un desplazamiento a su reloj nativo para sincronizarlo con el reloj maestro. Como los relojes son independientes, los desplazamientos deben ser actualizados regularmente. Todos los dispositivos que participan en la piconet están sincronizados en tiempo y saltos con el canal.

En una transmisión, cada paquete debe estar alineado con el inicio de un slot y puede tener una duración de hasta cinco timeslots. Durante la transmisión de un paquete la frecuencia es fija. Como ya se ha dicho, cada paquete es enviado en un salto de frecuencia RF distinta. Dicha frecuencia deriva del tiempo del reloj en cada instante. En el caso particular de paquetes mayores de una ranura temporal, todo el paquete se envía utilizando la frecuencia correspondiente al primer "slot" ocupado por el paquete. El paquete siguiente utiliza la frecuencia correspondiente al "slot" que ocupa; se olvida de la que sigue en la secuencia de saltos a la usada por el anterior paquete. Se puede observar el funcionamiento de esto en la figura 5.

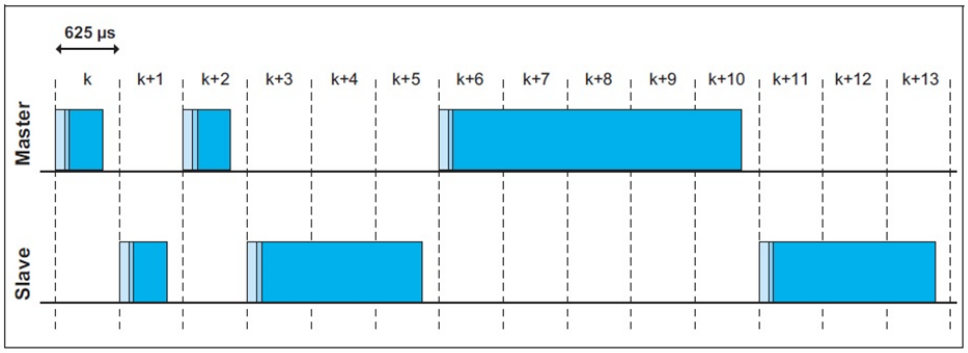

Figura 5. Paquetes multislot

Se usa un esquema TDD donde el maestro y el esclavo transmiten alternativamente. El inicio del paquete debe estar alineado con el inicio de la ranura. Los paquetes se pueden extender hasta cinco ranuras de tiempo.

2.3.3. Enlace físico

Un enlace físico representa una conexión de banda base entre dispositivos. Un enlace físico siempre está asociado con exactamente un canal físico. Los enlaces físicos tienen propiedades comunes que se aplican a todos los transportes lógicos en el enlace físico.

Las propiedades comunes de los enlaces físicos son:

- Control de potencia
- Supervisión del enlace
- Cifrado
- Cambio en la velocidad de datos impulsada por la calidad del canal
- Control de paquete de múltiples ranuras y para enlaces físicos en el canal físico de piconet adaptado:
- Mapa de canales AFH

El enlace físico Connectionless Slave Broadcast está asociado con el canal físico de piconet adaptado BR / EDR, un único transporte lógico y no admite el protocolo Link Manager.

Transportes lógicos

Entre maestro y esclavo (s), diferentes tipos de transporte lógico pueden ser establecidos. Se han definido cinco transportes lógicos:

- Transporte lógico sincrónico orientado a la conexión (SCO)
- Transporte lógico de conexión síncrona extendida (eSCO)

- Transporte lógico asíncrono orientado a la conexión (ACL)
- Transporte lógico de Active Slave Broadcast (ASB)
- Transporte lógico de conexión esclava sin conexión (CSB)

Los transportes lógicos sincrónicos son transportes lógicos punto a punto entre un maestro y un único esclavo en la piconet. Los transportes lógicos sincrónicos normalmente admiten información limitada en el tiempo como voz o datos síncronos generales. El maestro mantiene los transportes lógicos sincrónicos mediante el uso de ranuras reservadas a intervalos regulares. Además de las ranuras reservadas, el transporte lógico de eSCO puede tener una ventana de retransmisión después de las ranuras reservadas.

El transporte lógico de ACL también es un transporte lógico punto a punto entre el maestro y un esclavo. En las ranuras no reservadas para transporte (s) lógico (s) síncrono (s), el maestro puede establecer un transporte lógico de ACL por ranura para cualquier esclavo, incluidos los esclavos que ya participan en un transporte lógico síncrono.

El transporte lógico ASB es utilizado por un maestro para comunicarse con esclavos activos.

El transporte lógico CSB es utilizado por un maestro para enviar datos de transmisión de perfil a cero o más esclavos.

2.3.4. Formato de los paquetes banda base

En este apartado se explican los distintos formatos de paquete Bluetooth incluyendo los tipos de paquetes banda base para un canal ACL.

2.3.4.1. Velocidad básica (BR)

El formato de paquete general de los paquetes de velocidad básica se muestra en la figura número 6. Cada paquete consta de 3 entidades: el código de acceso, el encabezado y la carga útil. En la figura, se indica el número de bits por entidad.

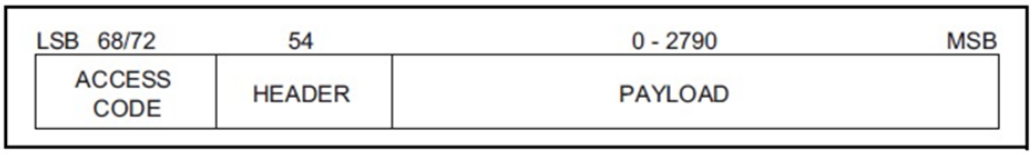

Figura 6. Formato de paquete de velocidad básica

El código de acceso es de 72 o 68 bits y el encabezado es de 54 bits. La carga útil oscila entre cero y un máximo de 2790 bits.

Se han definido diferentes tipos de paquetes. Un paquete puede consistir en:
- el código de acceso abreviado solamente.
- el código de acceso y el encabezado del paquete.
- el código de acceso, el encabezado del paquete y la carga útil.

Código de acceso

Cada paquete comienza con un código de acceso. Si sigue un encabezado de paquete, el código de acceso tiene 72 bits de longitud, de lo contrario, el código de acceso tiene 68 bits de

longitud y se conoce como código de acceso abreviado. El código de acceso abreviado no contiene un avance. Este código de acceso se usa para sincronización, compensación e identificación de compensación de CC. El código de acceso identifica todos los paquetes intercambiados en un canal físico: todos los paquetes enviados en el mismo canal físico están precedidos por el mismo código de acceso. En el receptor del dispositivo, se correlaciona con el código de acceso y se dispara cuando se excede un umbral. Esta señal de disparo se usa para determinar el tiempo de recepción.

El código de acceso abreviado se usa en paginación e indagación. En este caso, el código de acceso en sí se utiliza como un mensaje de señalización y ni un encabezado ni una carga están presentes.

El código de acceso consiste en un preámbulo, una palabra de sincronización y posiblemente un avance (tráiler) que depende del tipo de código de acceso.

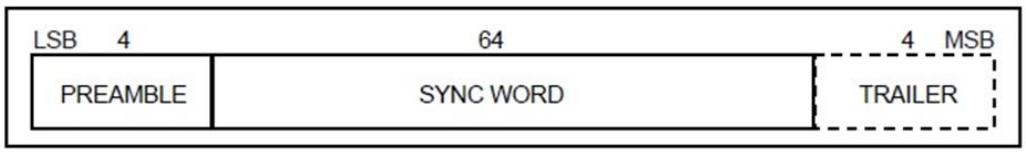

Figura 7. Formato del código de acceso

Cabecera de paquete

La cabecera contiene información de control de enlace (LC) y consta de 6 campos:

- **LT_ADDR**: dirección de transporte lógica de 3 bits
- **TIPO**: código de tipo de 4 bits
- **FLUJO**: control de flujo de 1 bit
- **ARQN**: indicación de confirmación de 1 bit
- **SEQN**: número de secuencia de 1 bit
- **HEC**: comprobación de error de encabezado de 8 bits

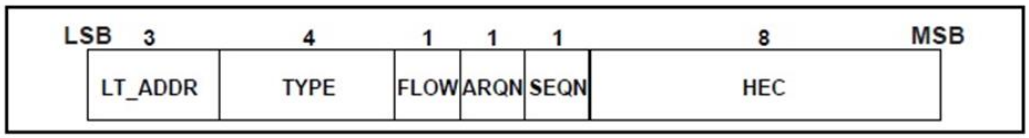

Figura 8. Formato de la cabecera de paquete

2.3.4.2. Velocidad de datos mejorada (EDR)

El formato general del paquete de velocidad de datos mejorada se muestra en la figura número 9. Cada paquete consta de 6 entidades: el código de acceso, la cabecera, el período de protección, la secuencia de sincronización, la carga útil de velocidad de datos mejorada y el remolque.

El código de acceso y el encabezado usan el mismo modo de modulación que los paquetes de frecuencia básica, mientras que la secuencia de sincronización, la carga útil de velocidad de datos mejorada y el trailer usan el modo de modulación de velocidad de datos mejorada. El tiempo de guardia permite la transición entre los modos de modulación.

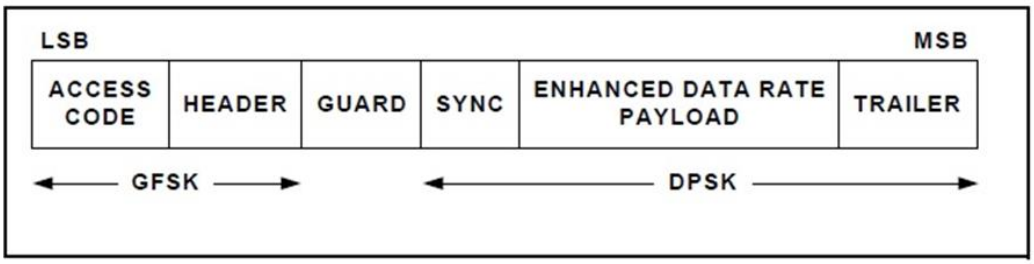

Figura 9. Formato de paquete de velocidad de datos mejorada

2.3.4.3. Tipos de paquetes banda base

A continuación se presentan unas tablas resumen de los diferentes tipos de paquetes banda base para un canal ACL y sus características principales.

Tipo	Cabecera de carga útil (bytes)	Carga útil de usuario (bytes)	FEC	CRC	Vel. Max. Simétrico (kb/s)	Vel. Max. Asimétrico (kb/s)	
						Bajada	Subida
DM1	1	0-17	2/3	Sí	108.8	108.8	108.8
DH1	1	0-27	No	Sí	172.8	172.8	172.8
DM3	2	0-121	2/3	Sí	258.1	387.2	54.4
DH3	2	0-183	No	Sí	390.4	585.6	86.4
DM5	2	0-224	2/3	Sí	286.7	477.8	36.3
DH5	2	0-339	No	Sí	433.9	723.2	57.6
AUX1	1	0-29	No	No	185.6	185.6	185.6
2-DH1	2	0-54	No	Si	345.6	345.6	345.6
2-DH3	2	0-367	No	Si	782.9	1174.4	172.8
2-DH5	2	0-679	No	Si	869.1	1448.5	115.2
3-DH1	2	0-83	No	Si	531.2	531.2	531.2
3-DH3	2	0-552	No	Si	1177.6	1766.4	235.6
3-DH5	2	0-1021	No	Si	1306.9	2178.1	177.1

Tabla 2. Tipos de paquete ACL

2.3.5. Corrección de errores

Hay tres esquemas de corrección de errores definidos para Bluetooth:

- Tasa de 1/3 FEC
- Tasa de 2/3 FEC
- Esquema ARQ para los datos

El objetivo del esquema de FEC en la carga de datos es reducir el número de retransmisiones. Sin embargo, en un entorno razonablemente libre de errores, FEC proporciona una sobrecarga innecesaria que reduce el rendimiento. Por lo tanto, las definiciones de paquete se han mantenido flexibles para usar FEC en la carga útil o no, resultando en los paquetes DM y DH para el transporte lógico ACL, paquetes HV para el transporte lógico SCO y paquetes EV para la lógica eSCO transporte. El encabezado del paquete siempre está protegido por un FEC de tasa 1/3, ya que contiene información de enlace valiosa y está diseñado para resistir más errores de bit.

2.3.6. Seguridad

El modelo de seguridad Bluetooth incluye cinco características de seguridad distintas: emparejamiento, vinculación, autenticación del dispositivo, encriptación e integridad del mensaje.

- Emparejamiento (Pairing): el proceso para crear una o más claves secretas compartidas
- Vinculación (Bonding): el acto de almacenar las claves creadas durante el emparejamiento para su uso en conexiones posteriores para formar un par de dispositivos confiables.
- Autenticación del dispositivo (Device authentication:): verificación de que los dos dispositivos tienen las mismas claves.
- Cifrado (Encryption): confidencialidad del mensaje
- Integridad del mensaje (Message integrity): protege contra falsificaciones de mensajes

2.3.7. Funcionamiento del controlador de enlace

Esta sección describe cómo se establece una piconet y cómo se pueden agregar y liberar dispositivos desde la piconet. Se definen varios estados de funcionamiento de los dispositivos para admitir estas funciones. Además, se discute la operación de varias piconets con uno o más miembros comunes, scatternet.

Vista general de los estados

La figura 10 muestra un diagrama de estado que ilustra los diferentes estados utilizados en el controlador de enlace. Hay dos estados principales: STANDBY y CONNECTION. Además, hay nueve subestados, página, escaneo de página, consulta, escaneo de búsqueda, tren de sincronización, escaneo de sincronización, respuesta maestra, respuesta de esclavo y respuesta de consulta. Tenga en cuenta que los subestados de respuesta maestra, respuesta esclava y respuesta de indagación no se muestran en la figura simplificada a continuación. Los subestados son estados intermedios que se utilizan para establecer conexiones y habilitar el descubrimiento de dispositivos. Para pasar de un estado o subestado a otro, se utilizan comandos del administrador de enlaces o se utilizan señales internas en el controlador de enlace.

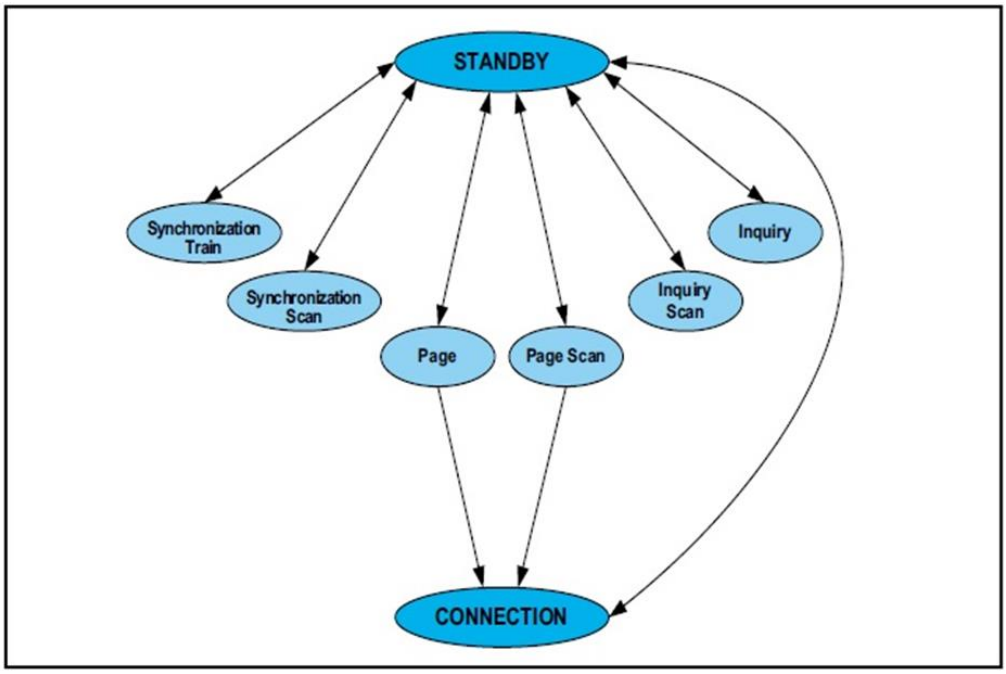

Figura 10. Vista general de los estados

2.4. Protocolo de gestión de enlace (LMP)

El protocolo de gestión de enlace (LMP) se utiliza para controlar y negociar todos los aspectos de la operación de la conexión Bluetooth entre dos dispositivos. Esto incluye la configuración y el control de transportes lógicos y enlaces lógicos, y para el control de enlaces físicos. [5]

El protocolo Link Manager se utiliza para comunicarse entre el enlace de gestión (LM) en los dos dispositivos. Todos los mensajes LMP se aplicarán únicamente al enlace físico y a los enlaces lógicos asociados y a los transportes lógicos entre los dispositivos emisor y receptor.
El protocolo está compuesto por una serie de mensajes que se transferirán sobre el enlace lógico ACL-C o ASB-C entre dos dispositivos. Los mensajes LMP deberán ser interpretados y actuados por el LM y no se propagará directamente a capas de protocolo superiores.

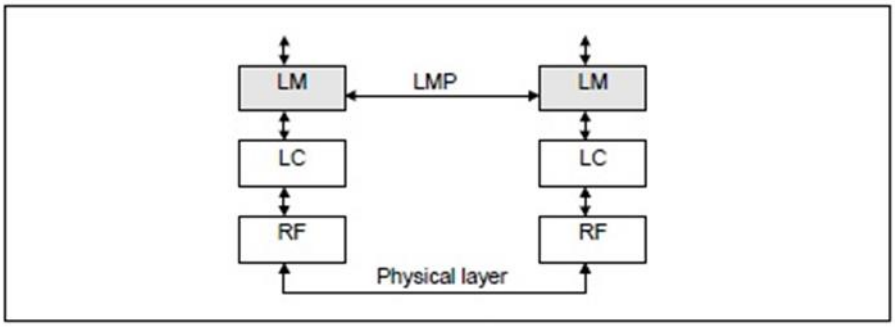

Figura 11. Capa de señalización del protocolo de gestión de enlace

Los mensajes LMP se transportarán en el transporte lógico de ACL predeterminado a menos que se especifique lo contrario. Los mensajes LMP tienen mayor prioridad que los datos de usuario, esto significa que si la gestión de enlace necesita enviar un mensaje, éste no debe ser retrasado por otro tráfico.

Solamente las retransmisiones de los paquetes del nivel de banda base pueden retrasar los mensajes LMP. Además, éstos no necesitan rutinas de reconocimiento ya que la capa banda base asegura un enlace fiable.

En resumen, Link Manager es responsable de la configuración del enlace y control de enlace, e incluye:

- Procedimientos para la creación y eliminación de una conexión
- Procedimientos de seguridad que incluyen autenticación, emparejamiento y encriptación
- Intercambio de información, p. acerca de la versión, funciones compatibles, etc.
- Control de modos de potencia y ciclos de servicio de la radio Bluetooth
- Calidad de servicio (QoS) con respecto a la asignación de ancho de banda

2.5. Interfaz de controlador de host (HCI)

El HCI proporciona un método de interfaz uniforme para acceder a las capacidades de un controlador Bluetooth banda base y a la gestión del enlace, además de acceso al hardware y a los registros de control. En otras palabras, brinda un método estándar para acceder a los recursos de banda base del sistema Bluetooth.

2.5.1. Capas inferiores de la pila Bluetooth

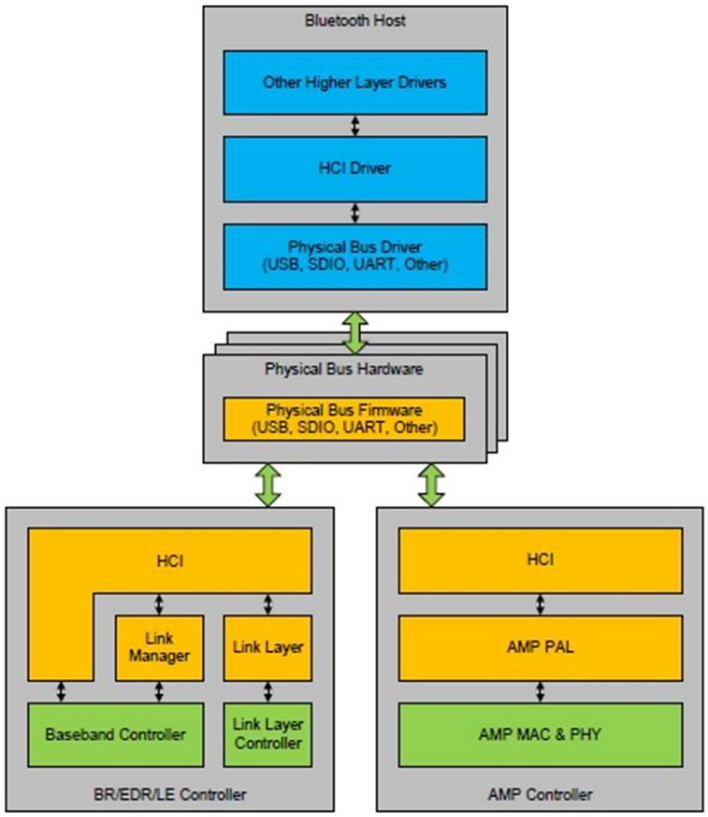

Figura 12. Vista general de las capas inferiores

Pueden existir varias capas entre el controlador HCI en el sistema Host y la capa HCI en los controladores. Estas capas intermedias, la capa de transporte del controlador de host, proporcionan la capacidad de transferir datos sin un conocimiento profundo de los datos.

La figura 13 muestra la ruta de una transferencia de datos de un dispositivo a otro. El controlador HCI en el host intercambia datos y comandos con el firmware HCI en el hardware Bluetooth. El controlador de la capa de transporte de control de host (es decir, el bus físico) proporciona a ambas capas de HCI la capacidad de intercambiar información entre sí.

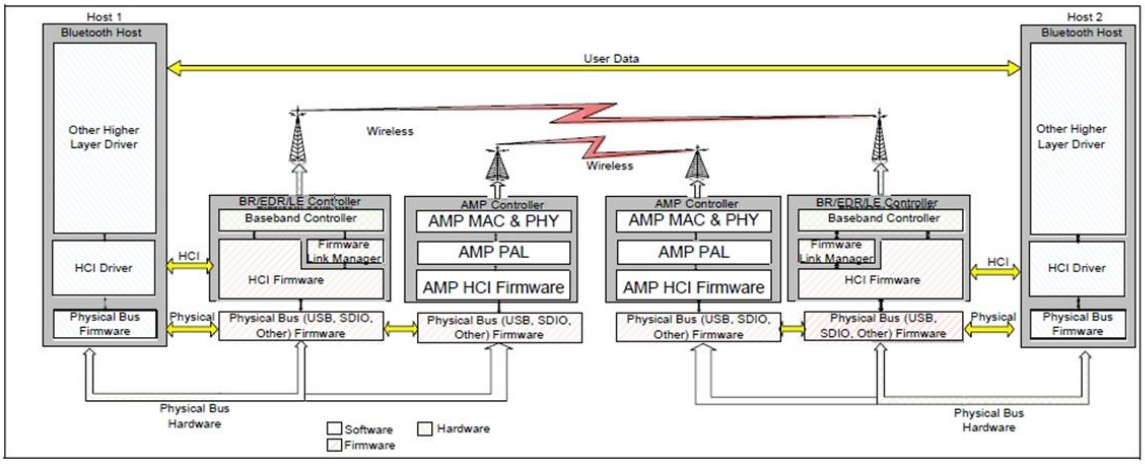

Figura 13. Descripción general de capas inferiores de software para transferir datos

El host recibirá notificaciones asincrónicas de eventos HCI independientemente de qué capa de transporte de controlador de host se utilice. Los eventos HCI se utilizan para notificar al host cuando ocurre algo. Cuando el Host descubre que ha ocurrido un evento, analizará el paquete de eventos recibidos para determinar qué evento ocurrió.

Un controlador BR / EDR / LE utiliza un buffer de comando compartido y control de flujo para BR / EDR y LE. Los almacenamientos intermedios de datos pueden compartirse entre BR / EDR y LE o pueden existir almacenamientos intermedios de datos separados para BR / EDR y LE. La configuración de un controlador se determina a través de HCI.

Los controladores LE usan un conjunto reducido de comandos y eventos HCI. Algunos comandos y eventos se reutilizan para múltiples tipos de controladores.

2.5.2. Visión general de la capa de transporte del controlador de host

La pila del controlador de host tiene una capa de transporte entre el driver de la interfaz de controlador de host y el host.

El objetivo principal de esta capa de transporte es la transparencia. El driver del controlador de host (que se conecta con el controlador) debe ser independiente de la tecnología de transporte subyacente. Además, el transporte no debería requerir ningún conocimiento de los datos que el driver del controlador de host pasa al controlador. Esto permite actualizar la interfaz lógica (HCI) o el controlador sin afectar la capa de transporte.

La interfaz lógica del controlador de host no considera la multiplexación/enrutamiento sobre las capas de transporte del controlador de host. El diseñador de host debe considerar esto al decidir cuál de las múltiples configuraciones de controlador admitirá.

2.6. Protocolo de control y adaptación de enlace lógico (L2CAP)

El “Logical Link Control and Adaptation Layer Protocol”, denominado como L2CAP, proporciona servicios de datos orientados a la conexión y no orientados a la conexión a protocolos de capa superior con capacidad de multiplexado de protocolo y operación de segmentación y reensamblaje. Un canal orientado a la conexión se usa para transportar datos punto a punto entre dos dispositivos, mientras que un canal no orientado a la conexión se utiliza para transmitir datos a múltiples receptores. La especificación L2CAP se define solo para los enlaces ACL y no se prevé la compatibilidad con los enlaces SCO. [5]

L2CAP permite protocolos y aplicaciones de nivel superior para transmitir y recibir paquetes de datos de capa superior (unidades de datos de servicio L2CAP, SDU) de hasta 64 kilobytes de longitud.

L2CAP también permite el control de flujo por canal y la retransmisión. La capa L2CAP proporciona canales lógicos, denominados canales L2CAP, que se multiplexan en uno o más enlaces lógicos.

Características del protocolo L2CAP

Los requisitos funcionales para L2CAP incluyen protocolo/canal de multiplexación, segmentación y reensamblado, control de flujo por canal y control de errores. L2CAP se encuentra sobre una capa inferior compuesta por uno de los siguientes:

1. Controlador BR / EDR y cero o más Controladores AMP.
2. Controlador BR / EDR / LE (compatible con BR / EDR y LE) y cero o más Controladores AMP.
3. Controlador LE (compatible solo con LE)

L2CAP interactúa con los protocolos de capa superior. En la figura siguiente se puede observar la arquitectura de bloques de L2CAP.

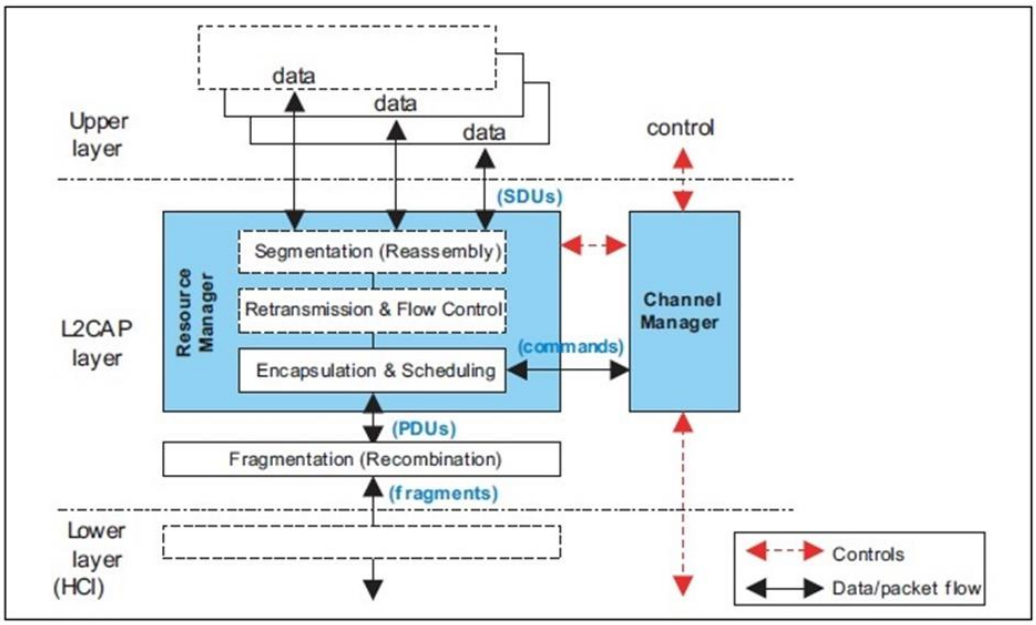

Figura 14. Arquitectura de bloques de L2CAP

Multiplexación de protocolos/canales

L2CAP admite la multiplexación en controladores individuales y en controladores múltiples. Un canal L2CAP debe operar sobre un controlador a la vez. Durante la configuración del canal, la capacidad de multiplexación del protocolo se usa para enrutar la conexión al protocolo correcto de la capa superior. Para la transferencia de datos, se necesita multiplexación de canales lógicos para distinguir entre múltiples entidades de capa superior. Puede haber más de una entidad de capa superior usando el mismo protocolo.

Segmentación y reensamblaje

Con el servicio de “frame relay” ofrecido por Resource Manager, la longitud de las tramas de transporte están controladas por las aplicaciones individuales que se ejecutan sobre L2CAP. Muchas aplicaciones multiplexadas están mejor servidas si L2CAP tiene control sobre la longitud de la PDU.

Esto proporciona los siguientes beneficios:

- La segmentación permitirá el entrelazado de unidades de datos de aplicación para satisfacer los requisitos de latencia.
- La administración de memoria y buffer es más fácil cuando L2CAP controla el tamaño del paquete.
- La corrección de errores por retransmisión puede hacerse más eficiente.
- La cantidad de datos que se destruyen cuando una PDU L2CAP es corrupta o perdida, se puede hacer más pequeña que la unidad de datos de la aplicación.
- La aplicación está desacoplada de la segmentación requerida para mapear los paquetes de la aplicación en los paquetes de la capa inferior.

Control de flujo por canal L2CAP

Los controladores proporcionan control de flujo y errores para los datos que pasan por el aire y existe control de flujo HCI para los datos que pasan por un transporte HCI. Cuando varias secuencias de datos se ejecutan en el mismo controlador utilizando canales L2CAP separados, cada canal requiere control de flujo individual. Se proporciona un esquema de control de flujo basado en el esquema de ventana.

Control de errores y retransmisiones

Cuando los canales L2CAP se mueven de un controlador a otro, se pueden perder datos. Además, algunas aplicaciones requieren una tasa de error residual mucho más pequeña que la que algunos controladores pueden ofrecer. L2CAP proporciona comprobaciones de errores y retransmisiones de PDU L2CAP. La comprobación de errores en L2CAP protege contra errores debido a que los controladores aceptan falsamente paquetes que contienen errores pero pasan las comprobaciones de integridad basadas en el controlador. La verificación y la retransmisión de errores de L2CAP también protegen contra la pérdida de paquetes debido al enjuague por parte del controlador. El control de errores funciona junto con el control de flujo en el sentido de que el mecanismo de control de flujo acelerará las retransmisiones, así como las primeras transmisiones.

Soporte para Streaming

Las aplicaciones de transmisión como audio configuran un canal L2CAP con una velocidad de datos acordada y no desean mecanismos de control de flujo, incluidos los del controlador, para alterar el flujo de datos. Un tiempo de espera de descarga se utiliza para mantener el flujo de datos en el lado de transmisión. El modo de transmisión por secuencias se usa para evitar que HCI y el control de flujo basado en el controlador se apliquen en el lado de recepción.

Fragmentación y recombinación

Algunos controladores pueden tener capacidades de transmisión limitadas y pueden requerir tamaños de fragmentos diferentes a los creados por la segmentación L2CAP.
Por lo tanto, las capas debajo de L2CAP pueden fragmentar y recombinar PDUs L2CAP para crear fragmentos que se ajusten a las capacidades de cada capa. Durante la transmisión de una PDU L2CAP, se pueden producir muchos niveles diferentes de fragmentación y recombinación en ambos dispositivos pares.

El driver HCI o controlador puede fragmentar PDU L2CAP para cumplir con las restricciones de tamaño de paquete de un esquema de transporte de interfaz de controlador de host. Esto da como resultado cargas útiles del paquete de datos HCI que portan fragmentos de inicio y continuación de la PDU L2CAP. De manera similar, el controlador puede fragmentar PDU L2CAP para mapearlas en paquetes de controlador. Esto puede dar como resultado cargas útiles del paquete del controlador que portan fragmentos de inicio y continuación de la PDU L2CAP.

Cada capa de la pila de protocolos puede pasar fragmentos de diferentes tamaños de PDU L2CAP, y el tamaño de los fragmentos creados por una capa puede ser diferente en cada dispositivo. Sin embargo, la PDU está fragmentada dentro de la pila, la entidad receptora L2CAP todavía recombina los fragmentos para obtener la PDU L2CAP original.

Calidad de servicio

El proceso de establecimiento de la conexión L2CAP permite el intercambio de información sobre la calidad del servicio (QoS) esperada entre dos dispositivos Bluetooth. Cada implementación de L2CAP supervisa los recursos utilizados por el protocolo y garantiza que se cumplan los contratos de QoS.

2.7. Protocolo de descubrimiento de servicio (SDP)

El protocolo de descubrimiento de servicios (SDP) proporciona un medio para que las aplicaciones descubran qué servicios están disponibles y para determinar las características de esos servicios disponibles. Como ejemplo, un portátil quiere reproducir un archivo de audio en un altavoz inalámbrico Bluetooth. En primer lugar, enviará un mensaje de consulta para descubrir dispositivos Bluetooth cercanos. En el siguiente paso, se conecta a estos dispositivos para buscar los servicios proporcionados por estos dispositivos. Para reproducir audio, buscará un dispositivo que proporcione el servicio Perfil de distribución de audio avanzado (A2DP). Una vez que se encuentra dicho dispositivo, puede crear una conexión A2DP con ese dispositivo para reproducir el archivo. [5]

El conjunto de servicios disponibles cambia dinámicamente en función de la proximidad de radiofrecuencia de los dispositivos en movimiento, es cualitativamente diferente del descubrimiento de servicios en entornos tradicionales basados en red. El protocolo de descubrimiento de servicio definido en esta especificación está destinado a abordar las características únicas del entorno Bluetooth.

Arquitectura del cliente-servidor SDP

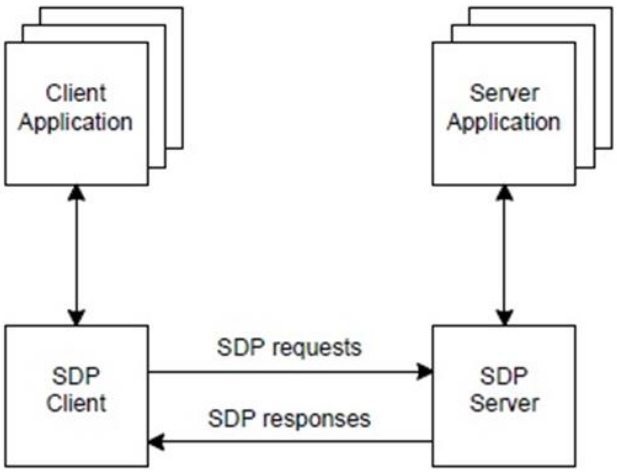

Figura 15. Interacción cliente-servidor SDP

El mecanismo de descubrimiento de servicios proporciona los medios para que las aplicaciones cliente descubran la existencia de servicios proporcionados por aplicaciones de servidor, así como los atributos de esos servicios. Los atributos de un servicio incluyen el tipo o clase de

servicio ofrecido y el mecanismo o la información de protocolo necesaria para utilizar el servicio.

SDP implica la comunicación entre un servidor SDP y un cliente SDP. El servidor mantiene una lista de registros de servicio que describen las características de los servicios asociados con el servidor. Cada registro de servicio contiene información sobre un solo servicio. Un cliente puede recuperar información de un registro de servicio mantenido por el servidor SDP emitiendo una solicitud SDP.

Si el cliente, o una aplicación asociada con el cliente, deciden usar un servicio, abre una conexión separada al proveedor del servicio para utilizar el servicio. SDP proporciona un mecanismo para descubrir servicios y sus atributos (incluidos los protocolos de acceso a servicios asociados), pero no proporciona un mecanismo para utilizar esos servicios (como la entrega de los protocolos de acceso al servicio).

Si hay varias aplicaciones en un dispositivo que brindan servicios, un servidor SDP puede actuar en nombre de esos proveedores de servicios para manejar las solicitudes de información sobre los servicios que brindan. Del mismo modo, las aplicaciones de múltiples clientes pueden utilizar un cliente SDP para consultar servidores en nombre de las aplicaciones del cliente.

El conjunto de servidores SDP que están disponibles para un cliente SDP cambiará dinámicamente en función de la proximidad de radiofrecuencia de los servidores con el cliente. Cuando un servidor está disponible, un cliente potencial debe ser notificado por un medio que no sea SDP para que el cliente pueda usar SDP para consultar al servidor sobre sus servicios.
De forma similar, cuando un servidor deja la proximidad o deja de estar disponible por algún motivo, no hay notificación explícita a través del protocolo de descubrimiento de servicio.
Sin embargo, el cliente puede usar SDP para sondear el servidor y puede deducir que el servidor no está disponible si ya no responde a las solicitudes.

2.8. Radio Frecuency Communication (RFCOMM)

El protocolo Bluetooth de Comunicación de radiofrecuencia (RFCOMM) es un conjunto simple de protocolos de transporte, ubicado en la parte superior del protocolo L2CAP, que emula los nueve circuitos de los puertos seriales RS-232 (ITU-T V.24). El protocolo RFCOMM admite hasta 60 conexiones simultáneas entre dos dispositivos Bluetooth. La cantidad de conexiones que se pueden usar simultáneamente en un dispositivo Bluetooth depende de la implementación. [6]

Recuerde que uno de los propósitos originales de la tecnología Bluetooth era reemplazar el cable serial y RFCOMM es el componente clave para permitir este reemplazo.

El perfil del puerto serie de Bluetooth (SPP) se basa en este protocolo. Muchas aplicaciones Bluetooth usan RFCOMM debido a su amplio soporte y la Interfaz de Programación de Aplicaciones (API) públicamente disponible en la mayoría de los sistemas operativos. RFCOMM admite hasta 60 conexiones simultáneas entre dos dispositivos Bluetooth, lo que permite que múltiples aplicaciones independientes se ejecuten e intercambien datos en paralelo. El protocolo RFCOMM es un protocolo adoptado y se basa en el estándar ETSI TS 07.10.

2.9. Object Exchange Protocol (OBEX)

El Object Exchange Protocol (OBEX) es un protocolo de comunicación que ayuda a los dispositivos a intercambiar un espectro más amplio de datos "de una manera estandarizada sensible a los recursos". El protocolo se adopta de la Asociación de datos de infrarrojos (IrDaA). OBEX utiliza el modelo cliente / servidor, en el que el dispositivo solicitante se considera el dispositivo cliente, y se aplica en muchos perfiles de Bluetooth para intercambiar (empujar / tirar) o sincronizar objetos de datos, cómo tarjetas de visita, notas, imágenes, archivos, calendarios, etc.

2.10. Perfiles Bluetooth

La interoperabilidad de aplicaciones en el sistema Bluetooth se logra mediante los perfiles de Bluetooth. Los perfiles definen las funciones y características requeridas de cada capa de la pila de protocolos desde la radio Bluetooth, banda base, hasta L2CAP, RFCOMM, OBEX, etc. Un perfil define las interacciones verticales entre las capas, así como las interacciones de igual a igual de capas específicas entre dispositivos. Un perfil proporciona información sobre cómo cada una de estas capas se unen para implementar un modelo de uso específico.

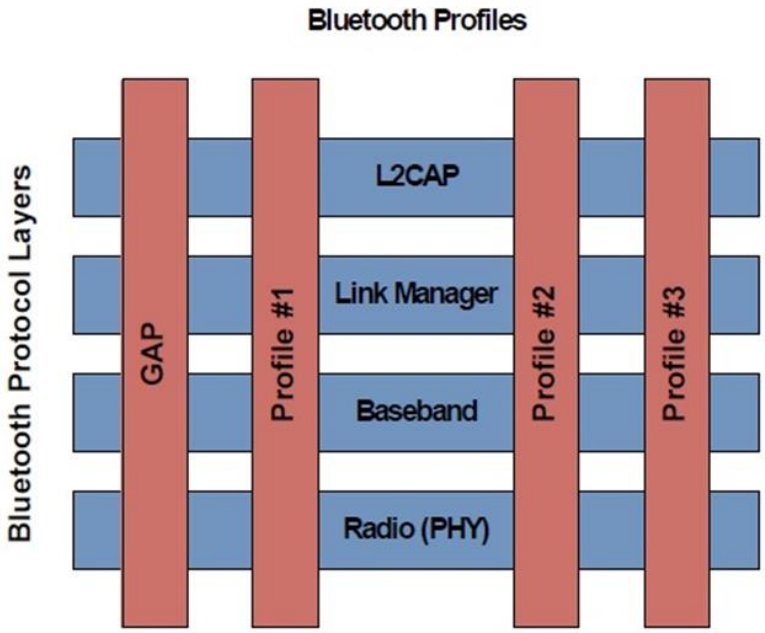

Figura 16. Perfiles Bluetooth

Además, los comportamientos de las aplicaciones y los formatos de datos también están definidos por perfil. Cuando dos dispositivos cumplen con todos los requisitos de un perfil Bluetooth, la interoperabilidad de una aplicación está habilitada.
Todos los perfiles describen los requisitos de descubrimiento de servicios necesarios para que los dispositivos se conecten, encuentren servicios de aplicaciones disponibles e información de conexión necesaria para hacer conexiones a nivel de aplicación.[5]

Los perfiles de Bluetooth se basan en la pila de protocolos de Bluetooth y, si bien las especificaciones de Bluetooth definen cómo funciona la tecnología, los perfiles definen cómo se usa. El propósito de los perfiles es garantizar que la tecnología sea fácil de usar y que se use correctamente. Los perfiles son esenciales en términos de interoperabilidad de aplicaciones, es

decir, ayudan a garantizar que la implementación de un proveedor funcionará correctamente con una implementación de otro vendedor.
Actualmente, hay más de 30 perfiles definidos en la especificación Bluetooth. El Generic Access Profile (GAP) es un perfil base que todos los dispositivos Bluetooth deben admitir. Un dispositivo típicamente admite varios perfiles al mismo tiempo. Qué perfiles soporta determina para qué aplicación es diseñado. Un auricular Bluetooth manos libres, por ejemplo, usaría un perfil de auricular (HSP), mientras que un controlador Nintendo Wii implementaría el perfil de dispositivo de interfaz humana (HID). Para que dos dispositivos Bluetooth sean compatibles, deben admitir los mismos perfiles.

Existen cuatro perfiles generales definidos, en los cuales están basados directamente algunos de los modelos de usuario más importantes y sus perfiles. Estos cuatro modelos son *Perfil Genérico de Acceso (GAP), Perfil de Puerto Serie, Perfil de Aplicación de Descubrimiento de Servicio (SDAP) y Perfil Genérico de Intercambio de Objetos (GOEP).*

La figura 17 muestra el esquema de los perfiles Bluetooth. En ella se pueden apreciar las jerarquías de los perfiles.

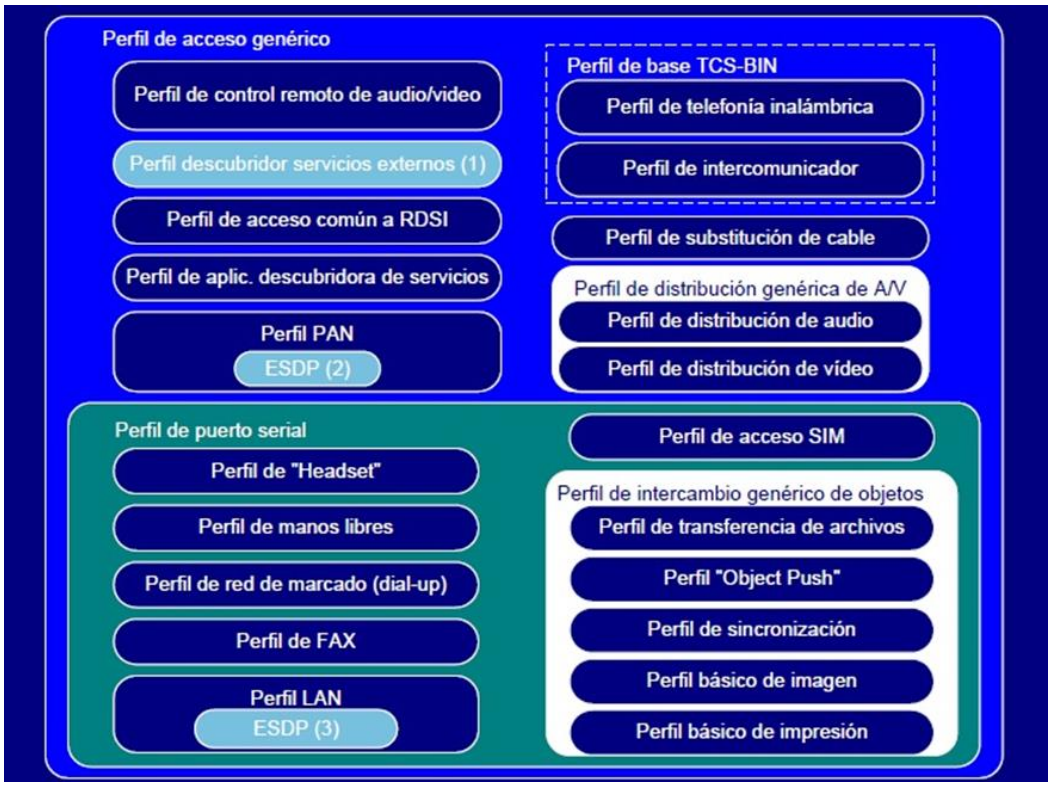

Figura 17. Jerarquías de los perfiles Bluetooth

A continuación se hace una breve descripción de estos y algunos otros perfiles Bluetooth.

- **Perfil Genérico de Acceso (GAP)**

  El sistema Bluetooth define un perfil base que implementan todos los dispositivos Bluetooth. Este perfil es el Perfil de acceso genérico (GAP), que define los requisitos básicos de un dispositivo Bluetooth. Por ejemplo, para BR / EDR, define un dispositivo Bluetooth para incluir radio, banda base, Link Manager, L2CAP y la funcionalidad de protocolo de descubrimiento de servicio; para LE, define la capa física, capa de enlace, L2CAP, administrador de seguridad, protocolo de atributo y perfil de atributo genérico. Esto une todas las capas para formar los requisitos básicos para un dispositivo

Bluetooth.

También describe los comportamientos y métodos para descubrimiento de dispositivos, establecimiento de conexiones, seguridad, autenticación, modelos de asociación y descubrimiento de servicios.

- **Perfil de Puerto Serie (SPP)**

Este perfil define los requerimientos para dispositivos Bluetooth, necesarios para establecer una conexión de cable serie emulada usando RFCOMM entre dos dispositivos similares. Este perfil solamente requiere soporte para paquetes de un slot. Esto significa que pueden alcanzarse flujos de datos de hasta 128 kbps. El soporte para flujos más altos es opcional. RFCOMM es usado para transportar los datos de usuario, señales de control de modem y comandos de configuración. El perfil de puerto serie es dependiente del GAP. [7]

La pila de protocolos de este perfil es la siguiente:

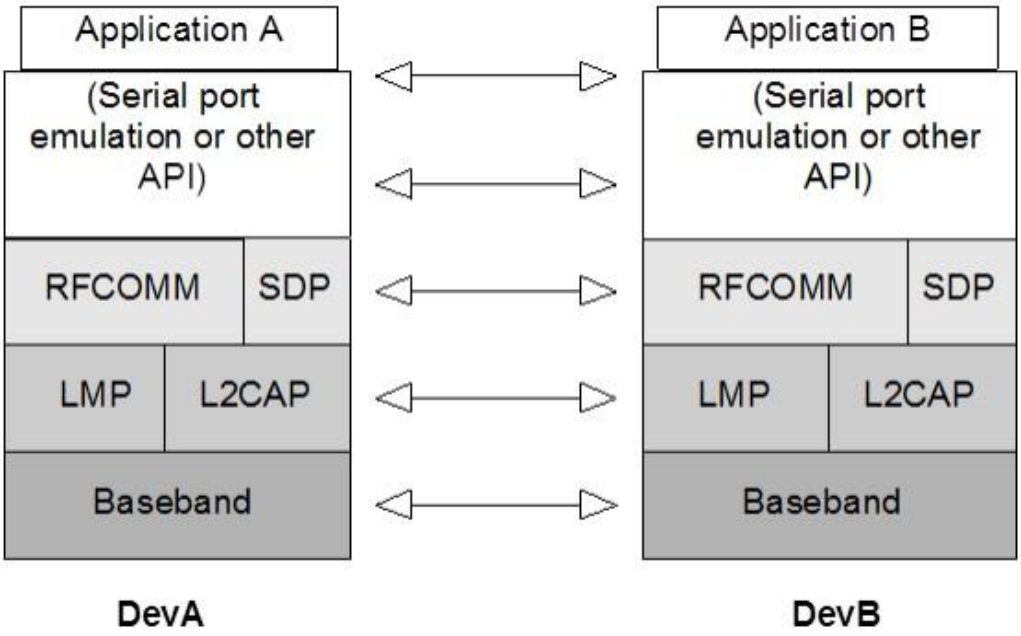

Figura 18. Pila de protocolos del Perfil de Puerto Serie (SPP)

- **Perfil de Aplicación de Descubrimiento de Servicio (SDAP)**

Este perfil define los protocolos y procedimientos para una aplicación en un dispositivo Bluetooth donde se desea descubrir y recuperar información relacionada con servicios localizados en otros dispositivos. El SDAP es dependiente del GAP.

- **Perfil Genérico de Intercambio de Objetos (GOEP)**

Este perfil define protocolos y procedimientos usados por aplicaciones para ofrecer características de intercambio de objetos. Los usos pueden ser, por ejemplo, sincronización, transferencia de archivos o modelo Object Push. Los dispositivos más comunes que usan este modelo son agendas electrónicas, PDAs, y teléfonos móviles. El GOEP es dependiente del SPP.[8]

Aquí vemos como este perfil utiliza el protocolo OBEX en el intercambio de objetos.

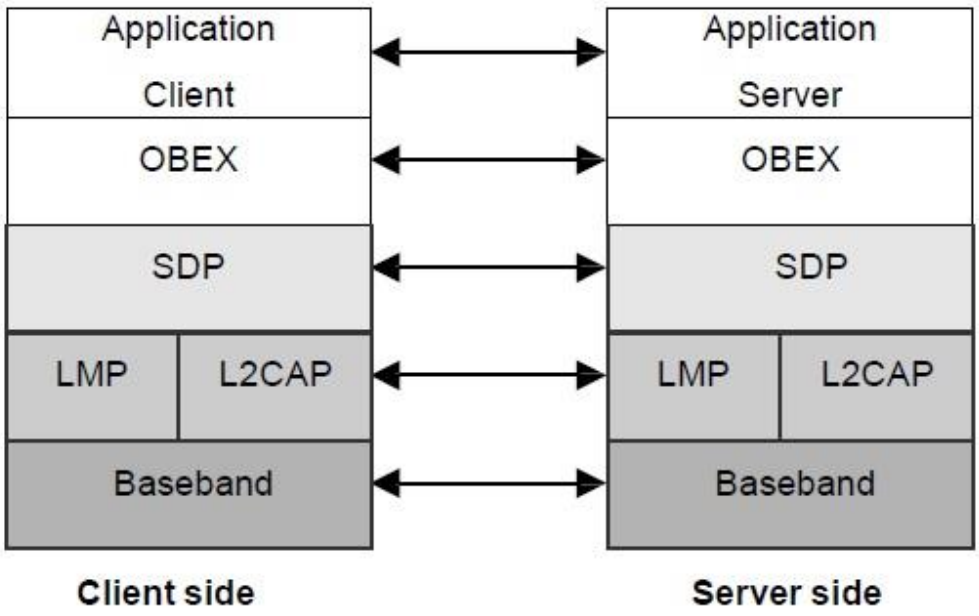

Figura 19. Pila de protocolos del GOEP

- **Perfil de Transferencia de ficheros (FTP)**

  El perfil de transferencia de archivos (FTP) define los requisitos para los protocolos y procedimientos que deben usar las aplicaciones que proporcionan el modelo de uso de transferencia de archivos. Este perfil utiliza el perfil genérico de intercambio de objetos (GOEP) para definir los requisitos de interoperabilidad para los protocolos necesarios para las aplicaciones.[9]

  Los escenarios cubiertos por este perfil son los siguientes:

  - Uso de un dispositivo Bluetooth para explorar un almacén de objetos (sistema de archivos) de otro dispositivo Bluetooth. La exploración implica visualizar objetos (archivos y carpetas) y navegar por la jerarquía de carpetas de otro dispositivo Bluetooth. Por ejemplo, un PC explorando el sistema de archivos de otro PC.

  - Un segundo uso es transferir objetos (archivos y carpetas) entre dos dispositivos Bluetooth. Por ejemplo, copiar archivos de un PC a otro PC.

  - Un tercer uso es para que un dispositivo Bluetooth manipule objetos (archivos y carpetas) en otro dispositivo Bluetooth. Esto incluye eliminar objetos y crear nuevas carpetas.

  El perfil se apoya en el perfil GOEP.

- **Perfil de manos libres (HFP)**

  Los ejemplos más comunes de los dispositivos que usan el perfil de manos libres son las unidades manos libres para automóviles que se usan junto con teléfonos celulares o auriculares inalámbricos portátiles.[10]

  El perfil define cómo dos dispositivos que soportan el perfil de manos libres interactuarán entre sí de punto a punto.

  Una implementación del perfil de manos libres normalmente permite que un auricular o una unidad de manos libres incorporada se conecte, de forma inalámbrica, a un teléfono móvil con el propósito de actuar como el mecanismo de entrada y salida de

audio del teléfono móvil y permitir que las funciones típicas de telefonía sean realizadas sin acceso al teléfono real.

El perfil de manos libres depende tanto del Perfil del Puerto Serie como del Perfil de Acceso Genérico.

- **Perfil de auriculares (HSP)**

Este perfil de auriculares define los protocolos y procedimientos que deben usar los dispositivos que requieren una conexión de audio de dúplex completo combinado con comandos mínimos de control del dispositivo. Los ejemplos más comunes de tales dispositivos son auriculares, personal computadoras, PDA y teléfonos móviles, aunque la mayoría de los teléfonos móviles preferirán usar un perfil más avanzado, como el perfil de manos libres.[11]

Los auriculares se pueden conectar de forma inalámbrica con el fin de actuar como el mecanismo de entrada y salida de audio del dispositivo, proporcionando audio dúplex completo. El auricular aumenta la movilidad del usuario mientras se mantiene la privacidad de la llamada.

El perfil de auriculares depende tanto del Perfil de Puerto Serie (SPP) como del Perfil de Acceso Genérico (GAP).

- **Perfil de distribución de audio avanzado (A2DP)**

El perfil de distribución de audio avanzado (A2DP) define los protocolos y procedimientos que realizan la distribución de contenido de audio de alta calidad en mono o estéreo en los canales de ACL. El término "audio avanzado", por lo tanto, se debe distinguir de "audio Bluetooth", que indica la distribución de voz de banda estrecha en canales SCO. [12]

Un caso de uso típico es la transmisión de contenido de música desde un reproductor de música estéreo a auriculares o altavoces. Los datos de audio se comprimen en un formato adecuado para un uso eficiente del ancho de banda limitado. La distribución de sonido envolvente no está incluida en el alcance de este perfil.

El A2DP se centra en la transmisión de audio, mientras que el perfil de distribución de video (VDP) especifica la transmisión de video. El soporte de ambos perfiles nos permite distribuir contenido de video acompañado de audio de alta calidad. El caso de uso de la transmisión de video y audio se describe en el VDP.

Tenga en cuenta también que el A2DP no incluye funciones de control remoto. Los dispositivos pueden admitir funciones de control remoto al implementar A2DP y el perfil de control como se describe, por ejemplo, en el escenario de uso del perfil de control remoto de audio / video.

El A2DP depende del Perfil de Acceso Genérico (GAP), así como del Perfil de distribución de audio / video genérico (GAVDP), que define los procedimientos necesarios para configurar una transmisión de audio / video.

- **Perfil de distribución de vídeo (VDP)**

El perfil de distribución de video (VDP) define los protocolos y procedimientos que realizan la distribución de contenido de video, utilizando los canales de ACL. Un caso típico de uso es la transmisión de contenido de video desde una cámara de observación a un monitor. Los datos de video están comprimidos en un formato específico para un uso eficiente del ancho de banda limitado. [13]

VDP se centra en la transmisión de video, mientras que el perfil de distribución de audio avanzado (A2DP) especifica la transmisión de audio de alta calidad. El soporte de ambos perfiles permite la distribución de contenido de video acompañado de audio de alta calidad.

VDP no incluye funciones de control remoto, y utiliza la misma arquitectura de transporte que A2DP. Los dispositivos pueden admitir funciones de control remoto en Bluetooth al implementar tanto el VDP como el perfil de control como se describe, por ejemplo, en el escenario de uso del perfil de control remoto de audio / video.

El VDP depende del Perfil de Acceso Genérico (GAP) y también del Perfil genérico de distribución de audio / video (GAVDP) que define los procedimientos necesarios para configurar una transmisión de audio / video.

- **Perfil LAN**

  El perfil LAN fue retirado el 3 de Junio de 2003 por lo que está actualmente obsoleto. El perfil LAN estaba apoyado en el perfil SPP como se muestra en la figura de las jerarquías, y a su vez utilizaba el protocolo RFCOMM en vez de apoyarse en el protocolo BNEP como hace el perfil PAN hoy en día.

- **Perfil PAN**

  Este es el perfil en el que se basará la parte práctica del proyecto, por ello está explicado en el capítulo 4 con mucho más detalle que los anteriores.

Capítulo 3. Protocolo de encapsulación de red Bluetooth (BNEP)

Bluetooth es una tecnología inalámbrica de corto alcance que opera en la banda ISM de 2,4 GHz. Muchos dispositivos como computadoras portátiles, teléfonos, PDA, electrodomésticos y otros dispositivos informáticos incorporan Bluetooth como parte del dispositivo. Los dispositivos habilitados para Bluetooth tendrán la capacidad de formar redes e intercambiar información. Para que estos dispositivos operen e intercambien información, se debe definir un formato de paquete común para encapsular los protocolos de red de la capa 3.

El Bluetooth Network Encapsulation Protocol o Protocolo de encapsulación de red de Bluetooth se encarga de encapsular paquetes de diferentes protocolos de red para que sean transportados directamente a través del protocolo de enlace lógico Bluetooth y L2CAP. El protocolo L2CAP proporciona la capa de enlace de datos de Bluetooth. [14]

El formato de paquete se basa en EthernetII / DIX Framing según lo definido por IEEE 802.3.

El perfil PAN (Personal Area Network) de Bluetooth describe cómo debe usarse BNEP para proporcionar capacidades de red a los dispositivos Bluetooth.

El protocolo BNEP debe ser compatible con los protocolos de red como IPv4, IPv6, IPX u otros existentes o emergentes. Otro requisito es que debe tener un bajo “overhead” para optimizar el ancho de banda y no empeorar la eficiencia.

3.1. Características generales

- Este protocolo está implementado sobre canales L2CAP orientados a la conexión.
- Bluetooth es considerado como un medio de transmisión al mismo nivel OSI que Ethernet, Token Ring, ATM, ...
- L2CAP es considerada la capa MAC (Media Access Control) de Bluetooth.
- BNEP especifica una MTU en L2CAP de 1691 bytes. Esta es además la MTU de L2CAP para BNEP, como IEEE 802.3.
- Todas las reglas de conectividad de red y topología definidas por 802.3 (como por ejemplo enrutamiento) son aplicadas consistentemente a Bluetooth.
- El espacio de direcciones de Bluetooth (BD_ADDR) está administrado por el IEEE, lo que significa que es posible construir un punto de acceso a la red como un puente entre dispositivos Bluetooth y una red Ethernet.

3.2. BNEP sobre L2CAP

Como se ha comentado anteriormente, el protocolo BNEP encapsula paquetes de diferentes protocolos de red para que sean transportados directamente a través del protocolo de enlace lógico Bluetooth y L2CAP. [14]

3.2.1. Pila de protocolos

Aquí vemos donde se encuentra situado el protocolo BNEP en la pila de protocolos Bluetooth.

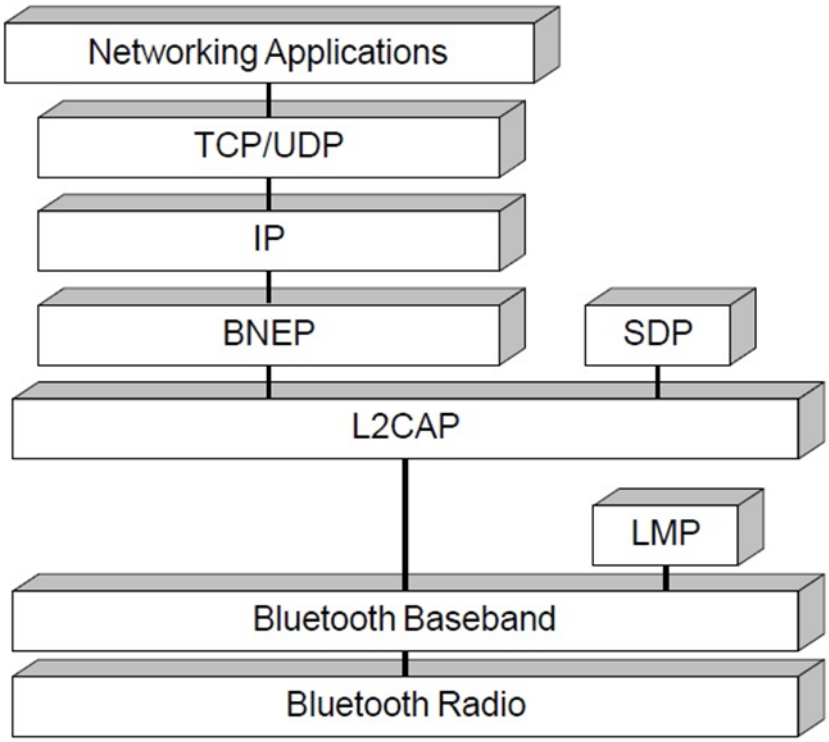

Figura 20. Pila de protocolos Bluetooth

3.2.2. Encapsulado de paquetes

Para transportar paquetes Ethernet, BNEP elimina la cabecera Ethernet y la reemplaza por su propia cabecera. El “payload” o carga útil permanece sin ningún cambio. El nuevo paquete (cabecera BNEP + payload) es encapsulado por L2CAP y enviado por Bluetooth.

La carga máxima que BNEP deberá aceptar de la capa superior es igual a la MTU L2CAP negociada (valor mínimo: 1691), menos 191 bytes (o 187 bytes si está presente un encabezado de etiqueta IEEE 802.1Q) reservada para encabezados BNEP. De esta forma, se puede asegurar que se reserva suficiente espacio en el búfer de cuadros para transmitir todos los BNEP. La carga útil mínima que BNEP deberá aceptar de la capa superior es cero; BNEP no está obligado a rellenar las cargas útiles con el tamaño mínimo de Ethernet (46 bytes).

La mínima MTU de 1691 fue seleccionada basándose en la máxima carga útil (payload) de un paquete Ethernet (1500 bytes) + la cabecera BNEP (15 bytes) + la cabecera L2CAP (4 bytes) + una posible extensión de la cabecera. Este tamaño es necesario para evitar la violación de cualquier regla que asuman los protocolos de capas superiores sobre la capa “EthernetII/DIX Framming like” de BNEP. 1691 = 5*339 (tamaño del DH5) – 4 (cabecera L2CAP).

El uso de BNEP para transportar paquetes Ethernet puede verse en la figura siguiente.

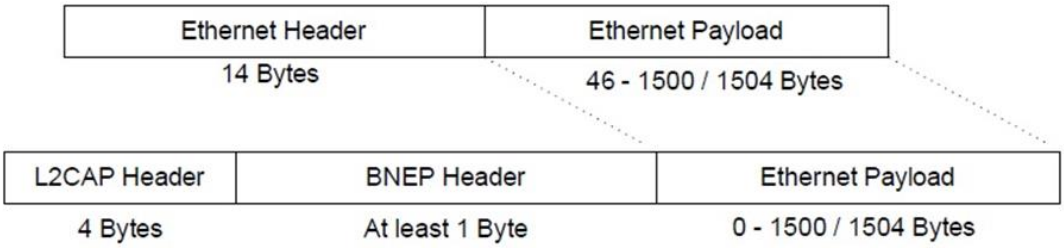

Figura 21. Encapsulado de paquetes BNEP

3.2.3. Formato de cabecera BNEP

Los dispositivos que soporten BNEP deben ser capaces de interpretar todos los tipos definidos de paquetes BNEP. Los dispositivos BNEP pueden transmitir opcionalmente las cabeceras BNEP comprimidas. Cualquier paquete que contenga un tipo de paquete reservado en la cabecera BNEP deberá ser descartado.

Todas las cabeceras BNEP siguen el formato mostrado en la figura siguiente.

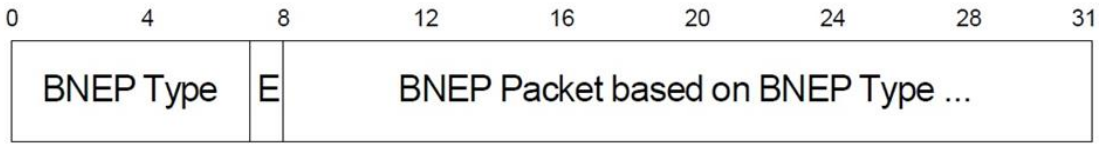

Figura 22. Formato de cabecera BNEP

A continuación, se analiza parte por parte la cabecera BNEP:

- **BNEP Type**: Este campo tiene un tamaño de siete bits e identifica el tipo de cabecera BNEP que contiene el paquete. Sus valores oscilan entre 0x00 – 0x7F y existen los siguientes tipos de paquete.

Value	BNEP Packet Type
0x00	BNEP_GENERAL_ETHERNET
0x01	BNEP_CONTROL
0x02	BNEP_COMPRESSED_ETHERNET
0x03	BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY
0x04	BNEP_COMPRESSED_ETHERNET_DEST_ONLY
0x05 – 0x7E	Reserved for future use
0x7F	Reserved for 802.2 LLC Packets for IEEE 802.15.1 WG

Figura 23. Tipos de paquete BNEP

- **Tipo de paquete BNEP_GENERAL_ETHERNET**. Este tipo de paquete se utiliza para transportar paquetes Ethernet desde y hacia redes Bluetooth. El paquete está conformado por una dirección destino, una dirección origen y el tipo de protocolo de red contenido en la carga útil (IPv4, IPv6, etc). Cualquiera de las direcciones ya sea origen o destino puede corresponder a una dirección Ethernet IEEE, si el origen o

destino es un dispositivo IEEE y no un dispositivo Bluetooth. Añade 14 bytes a la cabecera (6 bytes Destination Address + 6 bytes Source Address + 2 bytes Networking Protocol Type).

- **Tipo de paquete BNEP_CONTROL.** Este paquete se usa para el intercambio de información de control. En este tipo de paquete, toda la información de control está contenida en la cabecera del BNEP_CONTROL de tal manera que el campo de carga útil no contiene información alguna. Por el momento hay siete tipos de paquetes de control BNEP. El tipo de paquete de control es definido por el valor que se consigne en el campo tipo de control de BNEP; no se entra en detalle sobre cada tipo de paquete de control por salirse de los objetivos del proyecto, sin embargo, sí se nombrarán:

    - BNEP_CONTROL_COMMAND_NOT_UNDERSTOOD
    - BNEP_SETUP_CONNECTION_REQUEST_MSG
    - BNEP_SETUP_CONNECTION_RESPONSE_MSG
    - BNEP_FILTER_NET_TYPE_SET_MSG
    - BNEP_FILTER_NET_TYPE_RESPONSE_MSG
    - BNEP_FILTER_MULTI_ADDR_SET_MSG
    - BNEP_FILTER_MULTI_ADDR_RESPONSE_MSG

- **Tipo de paquete BNEP_COMPRESSED_ETHERNET**. Estos paquetes se utilizan para el transporte de paquetes Ethernet hacia o desde dispositivos con conexión directa a nivel de capa L2CAP usando BNEP. Debido a la existencia de una conexión L2CAP entre los dos dispositivos Bluetooth no es necesario incluir dentro del paquete las direcciones de origen y destino. Se debe anotar que las direcciones de multicast o broadcast no deben ser comprimidas. Añade 2 bytes a la cabecera (2 bytes Networking Protocol Type).

- **Tipo de Paquete BNEP_COMPRESSED_ETHERNET_SOURCE_ONLY.** Este tipo de paquete se usa para transportar paquetes Ethernet hacia un dispositivo que siempre será el destino final de los paquetes. Por esta razón los dispositivos no necesitan incluir la dirección destino en los paquetes, siendo ésta la misma dirección correspondiente al canal L2CAP sobre el cual se envían los paquetes. Se debe anotar que las direcciones de multicast o broadcast no deben ser comprimidas. Añade 8 bytes a la cabecera (6 bytes Source Address + 2 bytes Networking Protocol Type).

- **Tipo de Paquete BNEP_COMPRESSED_ETHERNET_DEST_ONLY.** Este paquete es usado para transportar paquetes desde un dispositivo que es la fuente del paquete. De esta manera los dispositivos no necesitan incluir la dirección de la fuente del paquete ya que esta fuente puede determinarse a partir de la conexión L2CAP. Añade 8 bytes a la cabecera (6 bytes Destination Address + 2 bytes Networking Protocol Type).

➢ **Bandera de Extensión (E):** Corresponde a una bandera de extensión de un bit de longitud que indica si existe una o más cabeceras de extensión entre la cabecera de BNEP y la carga útil. Si toma un valor de 0x1, entonces uno o más encabezados de extensión se ubican antes de la carga útil. Si el valor que tiene es 0x0, la carga útil sigue a la cabecera BNEP.

➢ **Paquete BNEP:** Depende del valor que se haya consignado en el campo del Tipo de BNEP.

3.2.4. Extensión de cabecera

Las cabeceras extendidas se usan como opciones suplementarias a la cabecera BNEP. Uno o más encabezados de extensión pueden incluirse después del encabezado BNEP y antes de la carga útil BNEP poniendo el bit de bandera de extensión BNEP a 1.

El tamaño total de todas las extensiones y cargas no puede exceder los 1676 bytes ya que la mínima cabecera BNEP son 15 bytes (BNEP_GENERAL_ETHERNET) y excedería la mínima MTU de BNEP de 1691 bytes.

En la figura siguiente, vemos el formato de la extensión de cabecera BNEP:

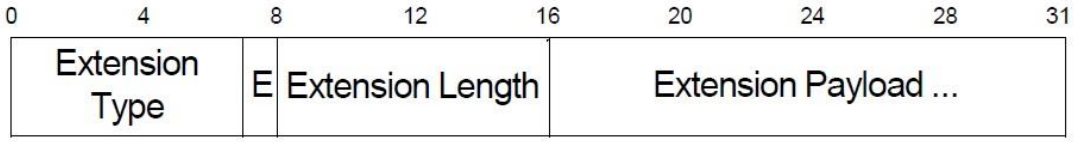

Figura 24. Extensión de cabecera BNEP

Se explica brevemente cada parte de la extensión de cabecera:

- **Extension Type:** Son 7 bits que indican el tipo de extensión. Actualmente sólo están definidas extensiones de control (0x00), pero se reservan el resto para futuros usos (0x01-0x7F).

- **Extension Flag (E):** Es un bit que indica si hay más extensiones de cabecera siguiendo a la actual. Si tiene el valor 1 es que siguen más extensiones a la actual y si tiene el valor 0 es que no hay y a continuación de esta extensión se encuentra el "payload" de BNEP.

- **Extension Length:** Es un byte que define el número de bytes que contiene la Extension Payload. Este byte no incluye los bytes que se usan para el Extension Type ni los Extension Length.

- **Extension Payload:** Basado en el Extension Type al igual que su tamaño.

Capítulo 4. Perfil PAN

El perfil de Personal Area Network (PAN) define un medio para permitir que los dispositivos Bluetooth participen en una red de área personal. La carga útil de los paquetes de Ethernet se transmite completamente intacta usando BNEP al intercambiar paquetes entre los dispositivos Bluetooth.[15]

Como ya se mencionó anteriormente, el perfil PAN es dependiente del Perfil de Acceso Genérico que ya fue descrito al hablar de los perfiles Bluetooth.

El perfil define cómo PAN es soportado en las siguientes situaciones:

- Red IP ad-hoc formada por dos o más dispositivos Bluetooth en una única piconet.
- Acceso a la red para uno o más dispositivos Bluetooth.

Sin embargo, el perfil PAN no cubre los siguientes temas:

- Formación automática de la red.
- Redes ad-hoc donde están involucradas múltiples piconets.
- Acceso indirecto a un punto de acceso a la red (NAP) a través de uno o más dispositivos Bluetooth intermedios.
- Grupo de redes ad-hoc conectadas a puntos de acceso a la red.
- Calidad de servicio (QoS).

4.1. Características generales

A continuación se enumeran las principales características de este perfil:

- Define redes personales dinámicas ad-hoc basadas en IP.

- Independiente del sistema operativo, del idioma y del dispositivo.

- Proporciona soporte para los protocolos de red más comunes como IPv4 e IPv6. Para otros protocolos el soporte puede o no proporcionarse.

- Brinda soporte para puntos de acceso donde la red podría ser una LAN corporativa, redes celulares u otras redes de datos.

- Acomoda los pocos recursos disponibles para dispositivos pequeños en cuanto a memoria, potencia de procesamiento e interfaces de usuario.

- El tráfico puede originarse desde cualquier dispositivo conectado a la red y puede estar destinado a cualquier otro dispositivo conectado a la red. Cualquier medio de transporte adecuado puede estar involucrado en la ruta del tráfico, por ejemplo, Bluetooth, Ethernet, Token Ring, PSTN, ISDN, ATM, GSM, etc.

4.2. Descripción general del perfil

Como se ha comentado anteriormente, el perfil PAN define un medio para permitir que los dispositivos Bluetooth participen en una red de área personal. A continuación se analizan los diferentes escenarios posibles.[15]

4.2.1. Escenarios

Este perfil aparece en tres escenarios diferentes. Cada uno de ellos tiene una arquitectura de red única y requisitos de red únicos, pero todos son varias combinaciones de una PAN.

- **Puntos de acceso a la red (NAP).** Un punto de acceso a la red es una unidad que contiene uno o más dispositivos de radio Bluetooth y actúa como puente, proxy o enrutador entre una red Bluetooth y alguna otra tecnología de red (10BASE-T, GSM, etc). Para un dispositivo conectado a un punto de acceso, la conexión radio y el controlador host del punto de acceso aparecen como una conexión de bus directa a un dispositivo de interfaz de red con acceso a la misma. Cada punto de acceso a la red puede permitir a uno o más dispositivos conectarse a su servicio, lo que puede incluir el acceso a todos los recursos compartidos de una LAN. Los puntos de acceso a la red deberán proporcionar acceso a otras redes a través de tecnologías como Ethernet, ISDN, Home PNA, Cable Modem y celular.

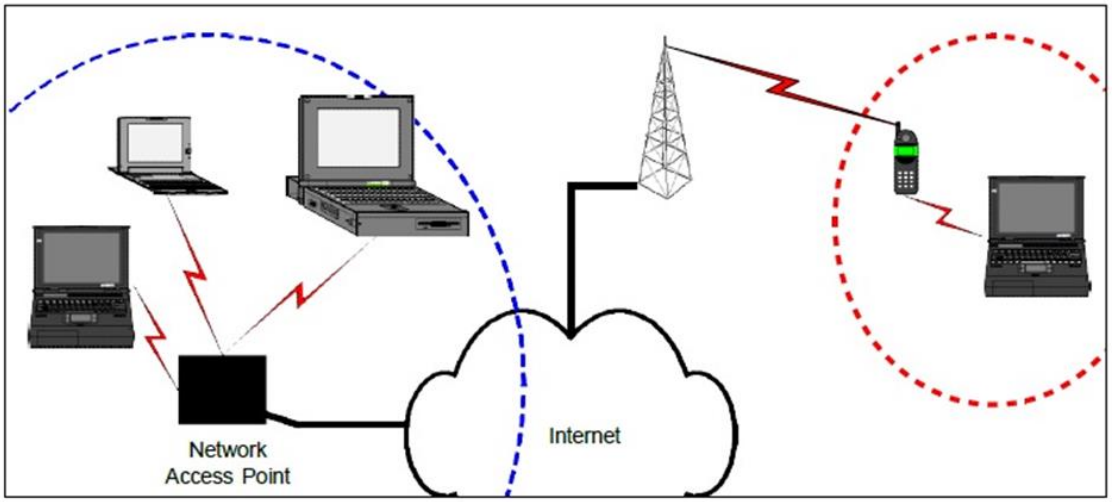

Figura 25. Ejemplo de dos tipos de NAP

- **Grupo de redes ad-hoc (GN).** Las redes ad-hoc permiten que los hosts móviles creen cooperativamente redes inalámbricas ad-hoc sin el uso de hardware o infraestructura de red adicional. El perfil PAN se centra en el escenario de una simple red personal ad-hoc que consiste en una única piconet Bluetooth con conexiones entre dos o más dispositivos Bluetooth.

  Una piconet consta de un dispositivo Bluetooth que funciona como maestro de piconet y se comunica con entre 1 y 7 dispositivos Bluetooth activos que funcionan como esclavos. Las comunicaciones en una piconet se realizan entre el maestro y los esclavos y bajo el control del maestro, ya sea de punto a punto o de punto a multipunto. Puede haber otros miembros de piconet no activos, que estén en modo de estacionamiento (park mode). La red ad-hoc es autónoma, es decir, forman una red sin necesidad de hardware de red externo adicional.

  A continuación, en la figura 26, vemos una red ad-hoc que consta de ocho dispositivos activos conectados en una única piconet.

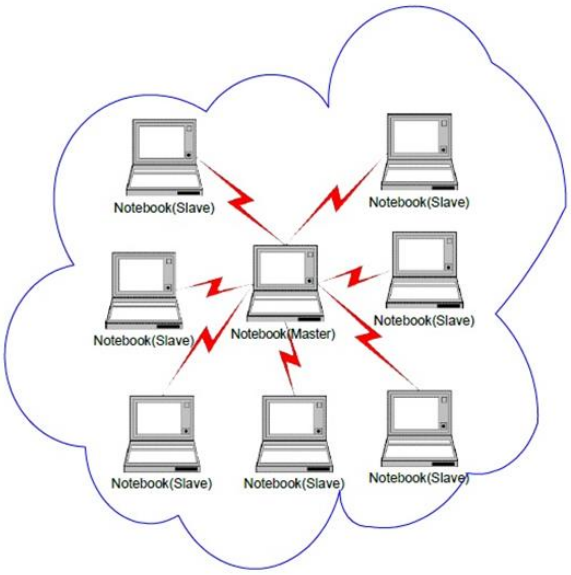

Figura 26. Red ad-hoc en una única piconet

- **PANU-PANU:** En este escenario, una conexión punto a punto entre dos PANU (Personal Area Network User), permite la comunicación directa solamente entre estos dos nodos.

Los servicios ofrecidos por el NAP cubren diferentes necesidades de red que los del GN. El NAP proporciona servicios de red a cada uno de los dispositivos Bluetooth conectados, mientras que el GN permite a dos o más dispositivos formar parte de una red Ad-hoc. Conectarse a un NAP o formar una GN entre varios dispositivos permite a las aplicaciones utilizar IP u otros protocolos de red sobre las conexiones Bluetooth.

Las conexiones entre dos PANU se usan únicamente para simular un cable cruzado entre dos nodos.

Como resumen de estos escenarios se incluye una ilustración básica de los tres tipos:

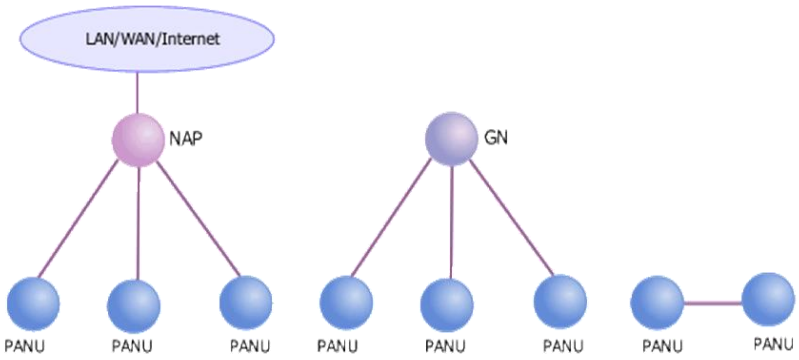

Figura 27. Tipos de escenarios posibles

4.2.2. Pila de protocolos

Las figuras siguientes muestran los protocolos y entidades utilizadas en cada uno de los tres escenarios definidos por el perfil PAN.

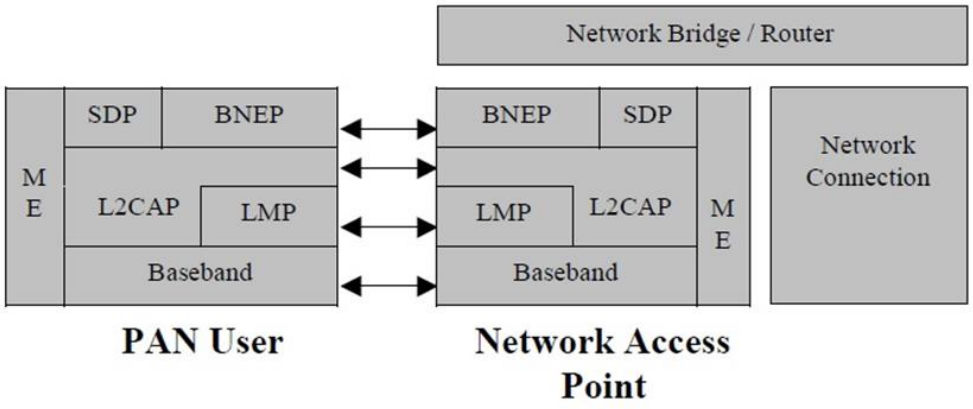

Figura 28. Pila de protocolos del escenario NAP

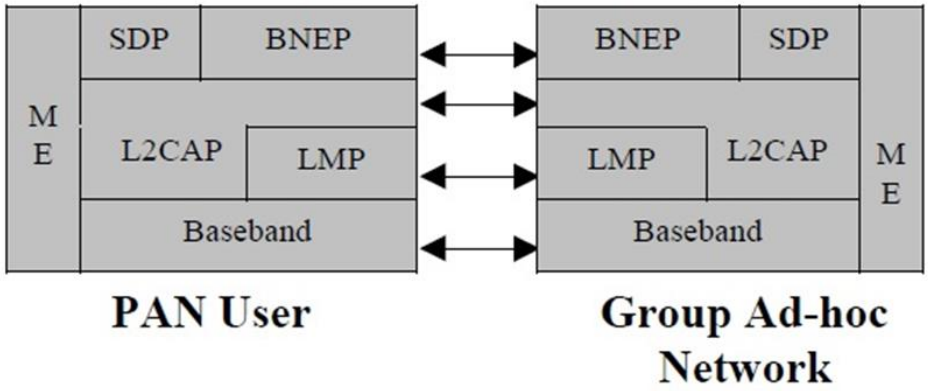

Figura 29. Pila de protocolos del escenario GN

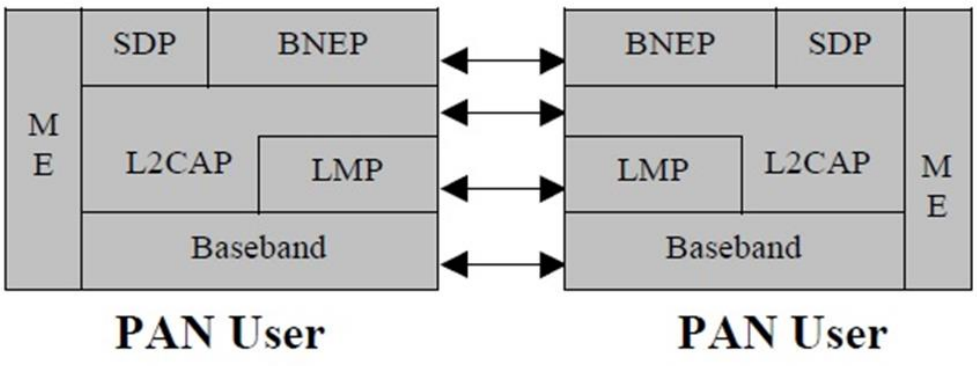

Figura 30. Pila de protocolos del escenario PANU-PANU

Ethernet Bridging se especifica en el estándar IEEE 802.1D. Este define un medio para mover paquetes de Ethernet entre varios medios y puertos. Solo una pequeña parte del estándar 802.1D es requerida por el perfil PAN.

La banda base, LMP y L2CAP son parte de los protocolos Bluetooth que residen en las capas 1 y 2 de OSI. SDP es el protocolo de descubrimiento de servicio Bluetooth.

ME es la entidad de gestión que coordina los procedimientos durante la inicialización, la configuración y la gestión de la conexión.

4.2.3. Configuraciones y roles

A continuación se puede ver una breve descripción de las diferentes configuraciones del perfil PAN:

- **Punto de Acceso a Red (NAP) y servicio NAP:** un dispositivo Bluetooth que soporta el servicio NAP, es un dispositivo que proporciona algunas de las características de un bridge Ethernet para soportar servicios de red. El dispositivo con el servicio NAP reenvía paquetes Ethernet entre los dispositivos Bluetooth conectados, conocidos como usuarios de red de área personal (PANU). El NAP y PANU intercambian datos usando el protocolo BNEP. El dispositivo con el servicio NAP tiene una conexión adicional de red a una arquitectura de red diferente a la cual se envían los paquetes Ethernet mediante un mecanismo de bridge de nivel 2 o un router de nivel 3. Estos dispositivos pueden necesitar funcionalidades adicionales cuando se utilizan redes adicionales como por ejemplo GPRS.

- **Grupo Ad-hoc Network (GN) y servicio GN:** un dispositivo Bluetooth que soporta el servicio GN es capaz de reenviar paquetes Ethernet a cada uno de los dispositivos Bluetooth conectados (PANUs) que lo necesiten. El GN y los PANU intercambian datos utilizando el protocolo BNEP. El GN no proporciona acceso a ninguna red adicional, en lugar de eso, permite a un grupo de dispositivos formar una red temporal e intercambiar información.

- **Usuarios PAN (PANU) y servicio PANU:** Este es el dispositivo Bluetooth que utiliza los servicios NAP o GN. El PANU actúa como cliente del NAP o del GN, y comunica directamente dos PANU.

Un dispositivo que soporte el perfil PAN puede ser capaz de proporcionar todos los servicios del perfil. Por ejemplo, un dispositivo podría soportar cualquier servicio ya sea NAP, GN y PANU. Si múltiples servicios son ofertados por un mismo dispositivo, entonces el usuario del perfil PAN, o la aplicación correspondiente, debe ser capaz de elegir cual es el que quiere usar. La selección del servicio está basada en los atributos de servicio que son hechos públicos mediante los Registros de Servicio de SDP.

La tabla que aparece a continuación explica las interacciones válidas entre los tres roles. El comportamiento de múltiples roles activos con conexiones en una radio no está definido, pero el tráfico para cada función no debe reenviarse entre roles.

Rol del aceptor		Rol del iniciador		
		NAP	**GN**	**PANU**
	NAP	NO	NO	SI
	GN	NO	NO	SI
	PANU	SI	SI	SI

Tabla 3. Interacciones válidas entre los tres roles del perfil PAN

4.2.4. Fundamentos del perfil

Los siguientes ejemplos ilustran cómo los roles NAP, GN y PANU interactúan en todas las combinaciones permitidas por el perfil PAN.

- **Ejemplo de NAP**

Se describe cómo un PANU descubre, conecta y utiliza un NAP y sus servicios de red. Este ejemplo proporciona un breve resumen de las interacciones típicas entre un NAP y un PANU.

1. El primer paso consiste en que el PANU descubra dentro de su radio de alcance un NAP disponible, que ofrezca el servicio de NAP al que el usuario o la aplicación desea conectarse. Para ello el PANU puede realizar una búsqueda banda base (“inquiry”) de equipos cercanos, y a continuación utilizar SDP para obtener el registro de NAP de aquellos dispositivos que admiten el rol de NAP. La lista de dispositivos devueltos por la orden de búsqueda puede ser filtrada según la clase de dispositivo para borrar a todos aquellos que no tengan activo el bit de red establecido en el campo de clase de servicio.

2. La elección del NAP con el que establecer una conexión puede basarse, entre otras razones, en los nombres de servicio extraídos de los registros de los NAP obtenidos por SDP. Cuando más de un NAP proporciona el servicio deseado, el usuario o una aplicación tendrá que elegir uno. Si no existe una conexión Bluetooth[1] con el dispositivo seleccionado, el PANU debe crearla. Si el NAP se encuentra en modo multiusuario se producirá un cambio maestro/esclavo para completar la conexión.

   [1] Se entiende por conexión Bluetooth, una conexión ACL establecida. Crear esto incluye todos los pasos intermedios relacionados con el establecimiento de la conexión ACL. Durante el establecimiento de la conexión, los dispositivos pueden realizar una autenticación mutua. Cualquiera de los dispositivos puede decidir el uso de encriptación en el enlace. La autenticación y la encriptación no son requeridas para la conexión. Además, durante el establecimiento de la conexión ACL puede realizarse un cambio maestro/esclavo.

3. Una vez que se ha creado una conexión ACL, el PANU inicia el establecimiento de un canal L2CAP para BNEP. A continuación se establece la conexión BNEP. El procedimiento de configuración de la conexión BNEP incluye los comandos de control BNEP requeridos.

4. El tráfico Ethernet puede ahora ir por el enlace. El PANU utiliza los servicios proporcionados por la red remota, como por la obtención de una dirección IP mediante DHCP. Si el PANU ya tiene una dirección IP para la conexión, como en el caso de moverse de un NAP a otro NAP, entonces el PANU verificará que dicha dirección IP siga siendo válida para esta nueva conexión. Otros servicios de red de interés también pueden ser utilizados por el PANU. El NAP debe reenviar los paquetes de Ethernet de manera apropiada a las PANU conectadas a través de la conexión de red NAP. Este es un comportamiento similar al de un puente de red.

5. Tanto el PANU como el NAP pueden terminar la conexión en cualquier momento.

- **Ejemplo de PANU iniciando una conexión a un GN**

Un PANU se conecta a un GN para crear una red ad-hoc con otros dispositivos Bluetooth. Este ejemplo proporciona un breve resumen de las interacciones típicas de un GN y un PANU.

1. El primer paso es encontrar otro dispositivo Bluetooth que se encuentre dentro del alcance radio y que esté proporcionando el servicio GN, utilizando consultas de banda base (“inquiry”) y búsquedas SDP.

2. Si no existe una conexión Bluetooth con el dispositivo que ofrece el servicio GN entonces el PANU debe iniciarla. Se requiere un cambio maestro-esclavo para completar la conexión si la GN está en modo multiusuario.

3. Una vez que se establece la conexión, el PANU puede crear un canal L2CAP para BNEP y usar los comandos de control BNEP para inicializar la conexión BNEP y configurar el filtrado de diferentes tipos de paquetes de red. Si el GN admite el filtrado, debe almacenar todos los filtros de tipo de paquete de red aceptados para cada conexión.

4. El tráfico de Ethernet ahora puede fluir a través del enlace. Es posible que los GN no proporcionen servicios de red y, por lo tanto, cada uno de los PANU realice varias tareas para operar sin estos servicios. El GN envía paquetes de Ethernet apropiadamente a los PANU conectados.

5. En cualquier momento pueden terminar la conexión tanto el PANU como el GN.

- **Ejemplo de un NAP/GN iniciando la conexión a un PANU**

Un NAP / GN se conecta a una PANU para crear una red ad-hoc con otros dispositivos Bluetooth. Esto solo podría ser posible si el PANU anuncia un registro de servicio PANU. Este ejemplo proporciona un breve resumen de las interacciones típicas de un NAP / GN y un PANU.

1. El primer paso es encontrar otro dispositivo Bluetooth que se encuentre dentro del alcance radio y que esté proporcionando el servicio PANU, utilizando consultas de banda base (“inquiry”) y búsquedas SDP.

2. Si no existe una conexión Bluetooth, el NAP / GN solicita una conexión Bluetooth con el dispositivo seleccionado con el servicio PANU. No se requiere de un cambio maestro-esclavo.

3. Una vez que se establece la conexión, el NAP / GN puede crear un canal L2CAP para BNEP y usar los comandos de control BNEP para inicializar la conexión BNEP. El PANU puede configurar el filtrado de diferentes tipos de paquetes de red. Si el NAP / GN admite el filtrado, debe almacenar todos los filtros de tipo de paquete de red aceptados para cada conexión.

4. El tráfico de Ethernet ahora puede ir a través del enlace. Los PANU realizan varias tareas para obtener una dirección IP y otros servicios de red. El NAP / GN debe reenviar todos los paquetes de Ethernet a cada una de las PANU conectadas.

5. En cualquier momento, tanto el PANU o el NAP / GN pueden terminar la conexión.

- **Ejemplo de una conexión PANU-PANU**

Un PANU se conecta a un PANU para simular una conexión de red de cable cruzado. Si se utiliza SDP antes de la configuración de la conexión, el PANU de destino debe anunciar un registro de servicio PANU. Este ejemplo proporciona un breve resumen de las interacciones típicas de dos PANU.

1. El primer paso es encontrar otro dispositivo Bluetooth que se encuentre dentro del alcance radio y que esté proporcionando el servicio PANU, utilizando consultas de banda base (“inquiry”) y búsquedas SDP.

2. Si no existe una conexión Bluetooth, el PANU solicita una conexión Bluetooth con el dispositivo seleccionado con el servicio PANU. No se requiere de un cambio maestro-esclavo.

3. Una vez que se establece la conexión, el PANU puede crear un canal L2CAP para BNEP. El PANU usa los comandos de control BNEP para inicializar la conexión BNEP.

4. El tráfico de Ethernet puede ir a través del enlace ahora. Ambas PANU realizarán varias tareas para obtener una dirección IP y otros servicios de red.

4.3. Modos de seguridad Bluetooth

Como el perfil de PAN depende del perfil de acceso genérico (GAP), las especificaciones con respecto a la seguridad del perfil de acceso genérico también son relevantes para el perfil de PAN. [15]
GAP especifica tres modos de seguridad, identificados aquí como los modos de seguridad Bluetooth:

- **No seguro**: un dispositivo no iniciará ningún procedimiento de seguridad.

- **Seguridad impuesta a nivel de servicio**: un dispositivo no inicia procedimientos de seguridad antes del establecimiento del canal en el nivel L2CAP.

- **Seguridad impuesta a nivel de enlace**: un dispositivo inicia los procedimientos de seguridad antes de que se complete la configuración del enlace en el nivel LMP.

4.3.1. Nivel de servicio de seguridad NAP/GN

El NAP / GN funciona en uno de los modos de seguridad de Bluetooth. Dentro del perfil PAN, el modo de seguridad 2 de Bluetooth (seguridad de nivel de servicio) se expande al modo de seguridad de nivel de servicio de perfil PAN, que incluye mecanismos de seguridad en el nivel de banda base Bluetooth, en un nivel de enlace superior (IEEE 802.1x) o en otra capa (IPSEC). Este modo está compuesto por los modos de autorización de perfil PAN y los modos de secreto de perfil PAN, ambos descritos más adelante.

Por ejemplo, supongamos que un NAP / GN está configurado en modo de seguridad reforzada de nivel de servicio de perfil PAN y ha establecido una conexión de banda base con una PANU. Supongamos ahora que la PANU desea conectarse al servicio NAP / GN, es decir, envía una L2CAP_ConnectReq para un canal BNEP. Entonces el NAP / GN deberá iniciar la seguridad de la conexión de acuerdo con los procedimientos de seguridad de Bluetooth. La seguridad de capa superior se inicia después del establecimiento de un canal L2CAP para BNEP. Los mecanismos de seguridad en diferentes niveles se pueden aplicar al mismo tiempo.

Además, la seguridad de los servicios a los que se accede a través de un NAP / GN puede ser soportada en el modo de seguridad de nivel de servicio de perfil PAN iniciando el Bluetooth y en los procedimientos de seguridad de nivel superior al conectarse al servicio particular. Esto está fuera del alcance del perfil de PAN.

4.3.1.1. Modos de autorización del perfil PAN

Los modos de autorización del perfil PAN especifican el nivel de autorización requerido para acceder a un PAN. El modo de autorización de perfil PAN es establecido por el NAP / GN, y se indica en el registro de servicio respectivo. El NAP / GN invoca los mecanismos de autenticación y autorización cuando se recibe un L2CAP_ConnectReq para un canal BNEP. El NAP / GN funciona en uno de los siguientes tres modos:

- **Open PAN:** No se requiere autorización ni autenticación para unirse a un PAN.
- **Autenticación:** es requerida por el NAP / GN antes de que el PANU esté registrado como miembro de la red de grupo ad-hoc. Si se utiliza la autenticación Bluetooth, se devuelve un L2CAP_ConnectRsp con el resultado conexión pendiente y la autenticación de estado pendiente. También se puede usar BNEP (802.1x) o autenticación de capa IP.
- **Autenticación y autorización:** Se requiere autorización y autenticación antes de que se complete el establecimiento de la conexión con PAN. Esto se puede hacer a nivel de Bluetooth, o en Ethernet (802.1x) o nivel de IP. A nivel de Bluetooth, la NAP / GN autoriza el establecimiento de un nuevo canal L2CAP. Un L2CAP_ConnectRsp con resultado conexión pendiente y estado autorización pendiente se devuelve. A nivel de BNEP o IP, se usa autenticación y autorización de ese nivel. Esto se inicia después de que se establece un canal L2CAP para BNEP.

Si un PANU no cumple con una solicitud de autenticación, o si un PANU no está autorizado por el NAP / GN, su canal L2CAP para BNEP deberá ser terminado por el NAP / GN. En caso de que se utilice la autenticación Bluetooth, se rechaza un mensaje L2CAP con resultado Connection Refused: se usa bloque de seguridad. Si un PANU se autentica

exitosamente a nivel Bluetooth y está autorizada por el NAP / GN para unirse al PAN, se devuelve un mensaje L2CAP con resultado de conexión exitosa.

4.3.1.2. Modos de confidencialidad del perfil PAN

El modo de confidencialidad del perfil PAN especifica el nivel de protección del tráfico dentro de la PAN. El nivel de confidencialidad es establecido por el NAP/GN. El PAN opera en uno de los siguientes modos:

- **Modo claro:** Lo que significa que no se aplica ninguna encriptación.
- **Modo encriptado:** Lo que significa que el cifrado se aplica en todas las comunicaciones dentro de la PAN. Esto puede ser a nivel de banda base o a nivel de BNEP / IP. Si se aplica el cifrado de banda base, deberá estar precedido por la autenticación de banda base. Entonces, el modo de autorización de perfil deberá ser el modo 2 o 3 que impone la autenticación de banda base.

Si un PANU no cumple con una solicitud de encriptación, su conexión con el PAN deberá terminar en el nivel L2CAP por el NAP / GN.

En caso de que se use el cifrado Bluetooth, se rechaza un mensaje L2CAP con resultado Connection Refused: se usa bloque de seguridad.
Si el cifrado se aplica a nivel de Bluetooth, y se deriva satisfactoriamente una clave de cifrado, se devuelve un mensaje de L2CAP con resultado de conexión exitosa.

En cualquier punto, el NAP / GN puede decidir cambiar el nivel de seguridad a un modo más seguro, es decir, desde el modo claro al modo encriptado. Un PANU que no cumpla con el cambio de modo deberá ser excluido del PAN.

4.3.2. Modos de seguridad PANU

Cualquier PANU que participe en una PAN puede exigir un cierto nivel de seguridad y, posteriormente, rechazar un menor nivel de seguridad si no se cumplen estas exigencias.

Esto da como resultado la terminación del canal de comunicación en el nivel relevante, es decir, relevante para el mecanismo de seguridad aplicado.

Por ejemplo, si la solicitud de autenticación Bluetooth no se cumple y la PANU está configurada en el modo de seguridad 3, la PANU terminará la conexión Bluetooth con el NAP / GN o la otra PANU. De forma similar, si la autenticación 802.1x falla, el canal L2CAP para BNEP deberá finalizar. Cuando el NAP / GN inicia el establecimiento de la conexión con el PANU, los procedimientos especificados anteriormente para NAP / GN también se aplican para este caso, con las funciones de NAP / GN y PANU invertidas. El modo de seguridad 2 (seguridad de nivel de servicio) se aplica cuando la PANU ha configurado requisitos de seguridad específicos para el canal L2CAP para BNEP (en este caso, el NAP / GN se conecta al "servicio PANU"). Si la PANU opera en modo de seguridad Bluetooth 2 o 3, la PANU debe darse cuenta de que los procedimientos de seguridad solo se aplican entre ese PANU en particular y el NAP / GN, y esa seguridad puede no aplicarse a otras conexiones. Un ejemplo de esta situación es si PANU y NAP están configurados para encriptar siempre el tráfico de banda base entre ellos.

4.3.3. Seguridad a nivel BNEP y superiores

La seguridad de banda base Bluetooth puede ser usada para proporcionar seguridad en el nivel de enlace donde opera. Al igual que otros protocolos de enlace no proporciona seguridad “end to end”.

Pueden utilizarse mecanismos de seguridad de niveles superiores a las capas de comunicación de Bluetooth, tales como VPN, IPSEC, TLS/WTLS, seguridad a nivel de aplicación, etc., para proporcionar la seguridad adecuada a una red específica. Estos mecanismos de seguridad son opcionales, no se exigen por el perfil PAN.

Para el perfil PAN, los mecanismos de seguridad brindan protección a los participantes en un PAN contra participantes no autorizados y escuchas de la comunicación Bluetooth de la capa de enlace. Sin embargo, los mecanismos de seguridad no protegen a los participantes del comportamiento malicioso de otros participantes en el mismo PAN, ni del comportamiento malicioso a través de una red externa conectada. Si se desea, se pueden aplicar mecanismos de seguridad para proteger la comunicación de un participante individual de un PAN para protegerse contra tales ataques. Ejemplos de tales mecanismos son IPSEC, TLS / WTLS y seguridad de nivel de aplicación.

Capítulo 5. Generación de la máquina virtual

El desarrollo práctico de este trabajo de fin de grado comienza con la instalación y configuración de una máquina virtual Linux, en concreto con la distribución Ubuntu. Se descargó el programa VMware Player, software gratuito que se utiliza para crear máquinas virtuales. Concretamente se usa la versión 17.10 que era la más reciente en el momento de la realización de este trabajo. Para ello, se descarga la siguiente imagen ISO de la página oficial de Ubuntu: "ubuntu-17.10.1-desktop-amd64" [16]. Se comprueba que el kernel es la versión 4.13.0-43-generic mediante el comando ***uname –r***

Una vez creada la máquina virtual, el comando ***bluetoothd –v*** permite conocer la versión del paquete BlueZ, que es la 5.46. El paquete BlueZ es la implementación de pila de protocolos Bluetooth. Se comprueba en su página oficial que la versión 5.46 es una de las versiones más recientes, incluida por defecto en la versión 17.10 de Ubuntu.

En la figura 31 se muestra el resultado de ejecutar el comando ***sudo dpkg –l | grep bluez*** que devuelve el conjunto de paquetes BlueZ instalados por defecto con sus correspondientes versiones.

```
local10@ubuntu:~$ sudo dpkg -l   | grep bluez
[sudo] password for local10:
ii  bluez                                        5.46-0ubuntu3
 amd64        Bluetooth tools and daemons
ii  bluez-cups                                   5.46-0ubuntu3
 amd64        Bluetooth printer driver for CUPS
ii  bluez-obexd                                  5.46-0ubuntu3
 amd64        bluez obex daemon
local10@ubuntu:~$
```

Figura 31. Paquetes BlueZ instalados por defecto

5.1. Configuración de la máquina virtual para el manejo de Bluetooth

En este apartado se describe el proceso de instalación de los paquetes y herramientas necesarios para el manejo y configuración de los dispositivos Bluetooth.

El paquete BlueZ-utils contiene la herramienta "bluetoothctl", que sirve para buscar, emparejar y conectar dispositivos desde la ventana del terminal de una forma muy sencilla. Este paquete no deja instalarlo porque está dentro del propio "bluez", y aunque no aparezca en la lista anterior como un paquete independiente, su funcionalidad de poder utilizar el "bluetoothctl" sí está disponible tal y como se muestra en la figura 32.

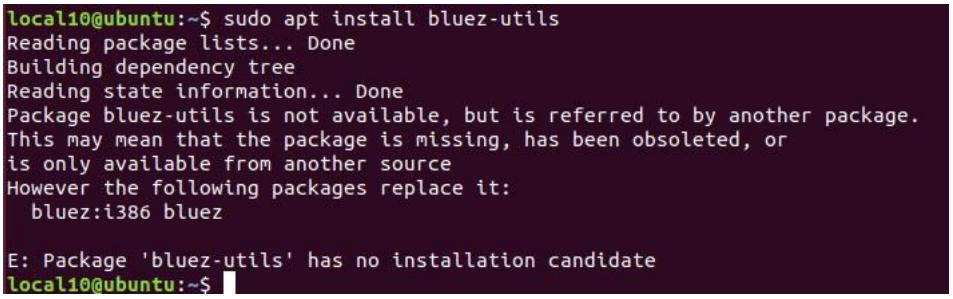

Figura 32. Instalación del paquete BlueZ-utils

A continuación, se instala el paquete BlueZ-hcidump que sirve para analizar los paquetes Bluetooth HCI desde una ventana de terminal. Esta función es un complemento al sniffer Wireshark a la hora de analizar los paquetes, y que se instalará posteriormente. Para instalar BlueZ-hcidump se ejecuta el comando ***sudo apt install bluez-hcidump*** en el terminal y ya se empezaría a instalar el paquete como se muestra en la figura 33.

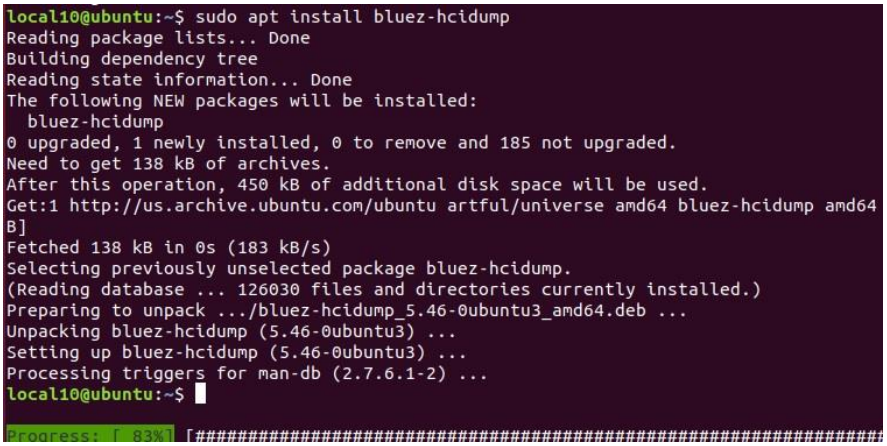

Figura 33. Instalación del paquete BlueZ-hcidump

Posteriormente se instala el paquete BlueZ-tools que incluye un conjunto de herramientas para administrar dispositivos Bluetooth para Linux. Se instala con el comando ***sudo apt install bluez-tools*** como se aprecia en la figura 34.

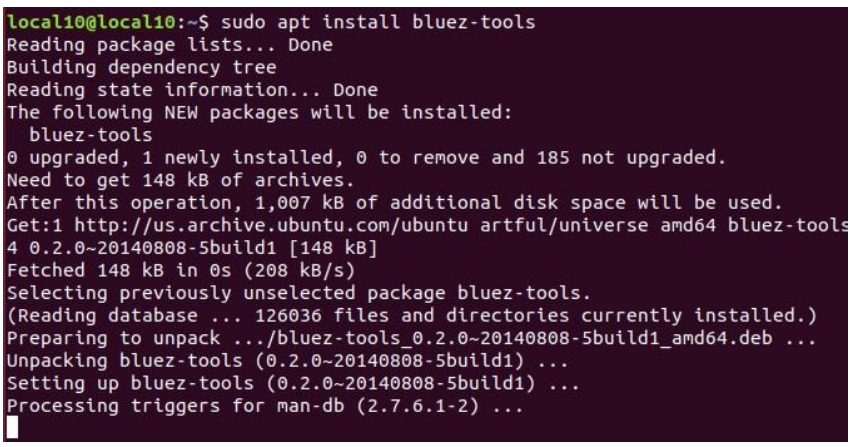

Figura 34. Instalación del paquete BlueZ-tools

Seguido se procede a instalar un paquete que tiene diversas utilidades que se tenían en las versiones anteriores de BlueZ, como el comando “pand” que sirve para iniciar el proceso del demonio PAN y que a partir de la versión 5 de BlueZ está obsoleto. Con él se puede, por

ejemplo, cambiar los roles de los dispositivos que es muy útil en el escenario GN del perfil PAN como se realizó en el proyecto “Despliegue de una red IP multisalto basada en Bluetooth”. [17] Este paquete es el BlueZ-compat. No se puede instalar directamente como los otros paquetes, ya que desde el terminal no se puede localizar. Por ello, se descargará en Internet desde la página oficial de Ubuntu. Está comprimido y se llama “bluez-compat_4.101-0ubuntu13.3_amd64.deb”.

Una vez descargado se necesita ejecutar un comando que permita descomprimir el paquete. Para ello instala “alien” mediante el comando sudo apt install alien y posteriormente se utiliza el propio comando “alien” para descomprimir el paquete Bluez-compat mediante ***sudo alien –i bluez-compat_4.101-0ubuntu13.3_amd64.deb*** desde el directorio donde se encuentra el paquete.

Por último se instalará el paquete “net-tools” que permitirá el uso de la utilidad “ifconfig” de Ubuntu. Con el comando “ifconfig” se puede conocer la información relativa a las interfaces de red, configurarlas o también habilitarlas y deshabilitarlas. Para instalar el paquete se ejecuta el comando ***sudo apt install net-tools*** como se muestra en la figura siguiente.

```
local10@local10:~$ sudo apt install net-tools
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 93 not upgraded.
Need to get 194 kB of archives.
After this operation, 803 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu artful/main amd64 net-tools amd64 1.60+g
Fetched 194 kB in 0s (237 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 174726 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20161116.90da8a0-1ubuntu1_amd64.deb ...
Unpacking net-tools (1.60+git20161116.90da8a0-1ubuntu1) ...
Processing triggers for man-db (2.7.6.1-2) ...
Setting up net-tools (1.60+git20161116.90da8a0-1ubuntu1) ...
```

Figura 35. Instalación del paquete net-tools

5.2. Instalación de Blueman

Otra herramienta que se va a instalar es el paquete Blueman, que es un administrador Bluetooth que contiene todas las funciones para su manejo. Proporciona un panel de configuración gráfico llamado blueman-manager. Esto permitirá buscar, emparejar, autenticar y conectar dispositivos desde su propio menú. Para instalarlo se ejecuta el comando ***sudo apt install blueman***

5.3. Instalación de Wireshark

Con objeto de poder monitorizar todo el tráfico Bluetooth, se instala Wireshark en las máquinas virtuales. Para ello se ejecuta el comando ***sudo apt install wireshark***. Una vez instalado el Wireshark, se observó que no aparecía ninguna interfaz Bluetooth sobre la que capturar como se muestra en la figura 36.

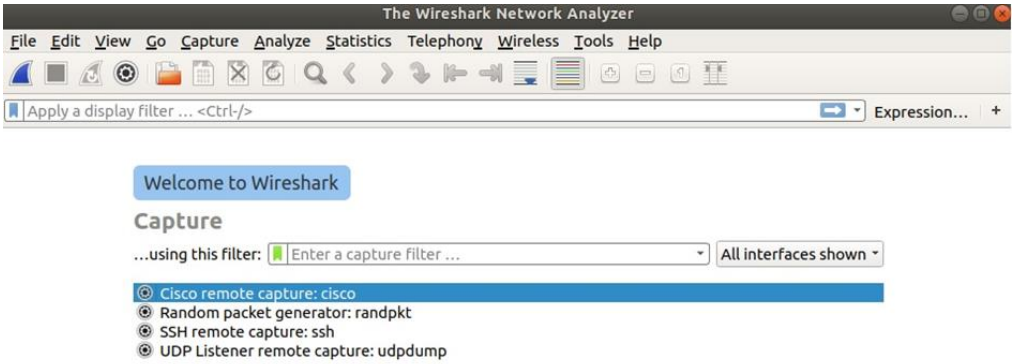

Figura 36. Wireshark sin interfaces

Para resolver este problema se encontró que era necesario ejecutar en el terminal los comandos descritos a continuación. [18]

sudo apt-get install wireshark libcap2-bin
sudo groupadd wireshark
sudo usermod -a -G wireshark $USER
sudo chgrp wireshark /usr/bin/dumpcap
sudo chmod 755 /usr/bin/dumpcap
sudo setcap cap_net_raw,cap_net_admin=eip /usr/bin/dumpcap

El primer comando sirve para instalar “libcap2-bin” que ya estaba instalado y el segundo comando para añadir el grupo wireshark que ya existe. Todo esto se muestra en la figura siguiente.

```
local10@local10:~$ sudo apt-get install wireshark libcap2-bin
Reading package lists... Done
Building dependency tree
Reading state information... Done
libcap2-bin is already the newest version (1:2.25-1.1).
wireshark is already the newest version (2.4.2-1).
0 upgraded, 0 newly installed, 0 to remove and 108 not upgraded.
local10@local10:~$ sudo groupadd wireshark
groupadd: group 'wireshark' already exists
local10@local10:~$ sudo usermod -a -G wireshark $USER
local10@local10:~$ sudo chgrp wireshark /usr/bin/dumpcap
local10@local10:~$ sudo chmod 755 /usr/bin/dumpcap
local10@local10:~$ sudo setcap cap_net_raw,cap_net_admin=eip /usr/bin/dumpcap
local10@local10:~$
```

Figura 37. Activación de las interfaces para capturar

Se vuelve a abrir Wireshark y se comprobará que aparecen todas las interfaces disponibles como se muestra en la figura 38, incluida la interfaz bluetooth0.

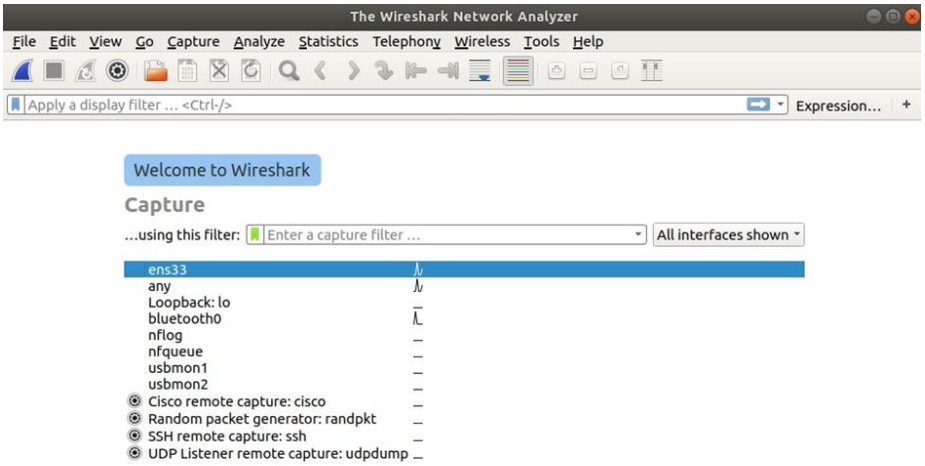

Figura 38. Wireshark con todas las interfaces para capturar

Capítulo 6. Despliegue de la red

La red que se va formar consiste en una piconet en la cual se tiene un dispositivo que realizará la función de NAP y varias PANUs que se conectarán con el propio NAP, que es el dispositivo que da salida a Internet.

En el laboratorio de telemática se han instalado tres máquinas virtuales, en tres ordenadores con sistema operativo Windows 7, procesador Intel(R) Core(TM) i7-3770 CPU @ 3,40 GHz y 8 Gb de memoria RAM. En cuanto al escenario PAN desplegado, consta de un dispositivo actuando como NAP en local7 y dos dispositivos actuando como PANU en local8 y local10. Conviene aclarar esto para que si se habla de un ordenador se sepa cuál es su función.

- LOCAL7: NAP
- LOCAL8: PANU
- LOCAL10: PANU

El esquema de la red desplegada es el de la siguiente figura:

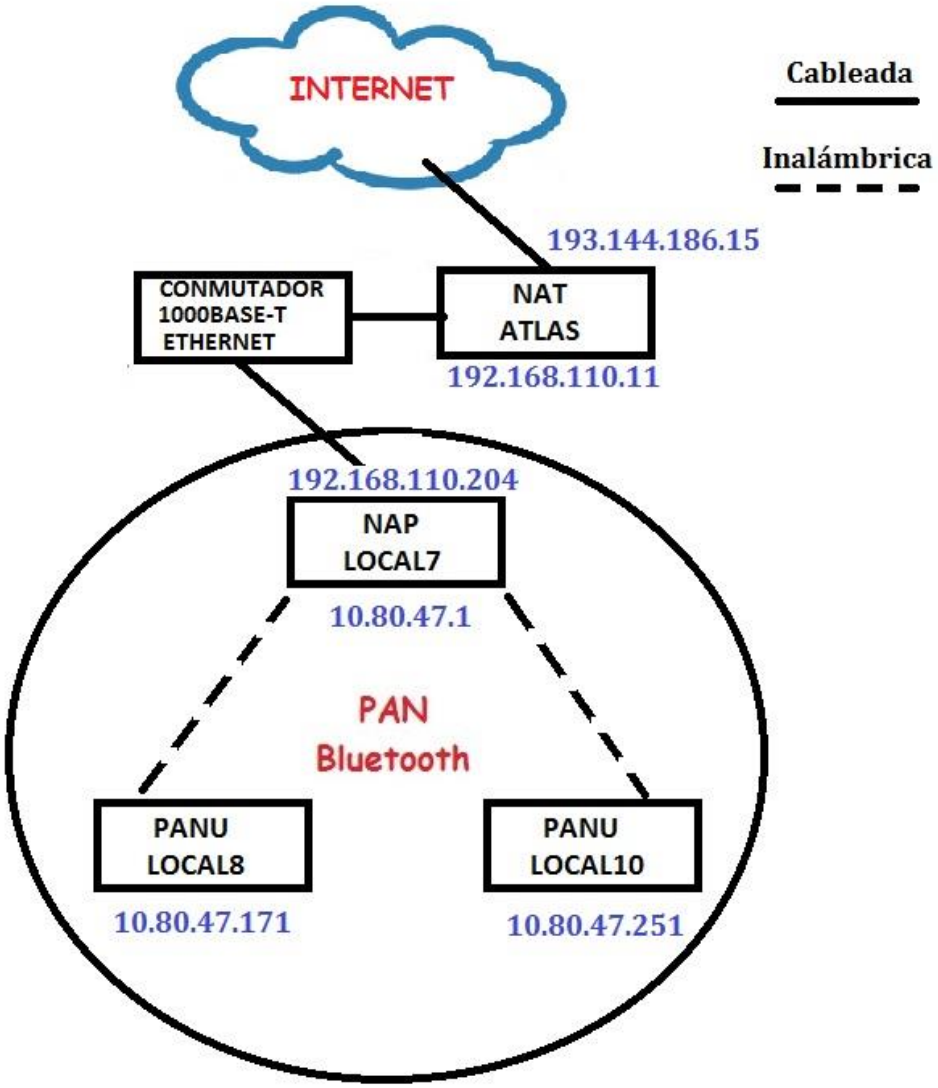

Figura 39. Escenario PAN desplegado

Mediante el comando “ifconfig” se muestran las características de las interfaces de red de los dispositivos. Se comienza por el NAP.

Figura 40. Salida del comando ifconfig en el NAP (local7)

Se aprecia que existen dos interfaces BNEP (bnep0 y bnep1) en el NAP, una por cada PANU conectado. Se muestra la dirección BD_ADDR del NAP (5c:f3:70:68:ee:e6) y los paquetes transmitidos y recibidos entre ambos dispositivos.

También se observa la interfaz pan1, que indica que hay una red de área personal. Este perfil dará acceso a la red a los dispositivos Bluetooth que forman la piconet. La dirección IP que adopta el NAP es de clase privada siendo 10.80.47.1. El NAP tendrá como Gateway del VMWare la dirección 192.168.72.128 para salir a través del NAT del laboratorio (Atlas) a Internet.

En las figuras siguientes se comprueban las direcciones IP privadas de los PANUs que en el caso de local8 es 10.80.47.171 y para local10 10.80.47.251. También aparecen las direcciones BD_ADDR de los dongles correspondientes (00:19:0e:16:e5:2e para local8 y 5c:f3:70:68:ee:e2 para local10). Se observa asimismo que no disponen de ninguna interfaz cableada para conectarse a la red. Las interfaces ens33 con direcciones 192.168.38.128 y 192.168.121.129 corresponden a las direcciones de la máquina virtual VMWare. En las siguientes figuras se muestran las interfaces de red de los PANUs.

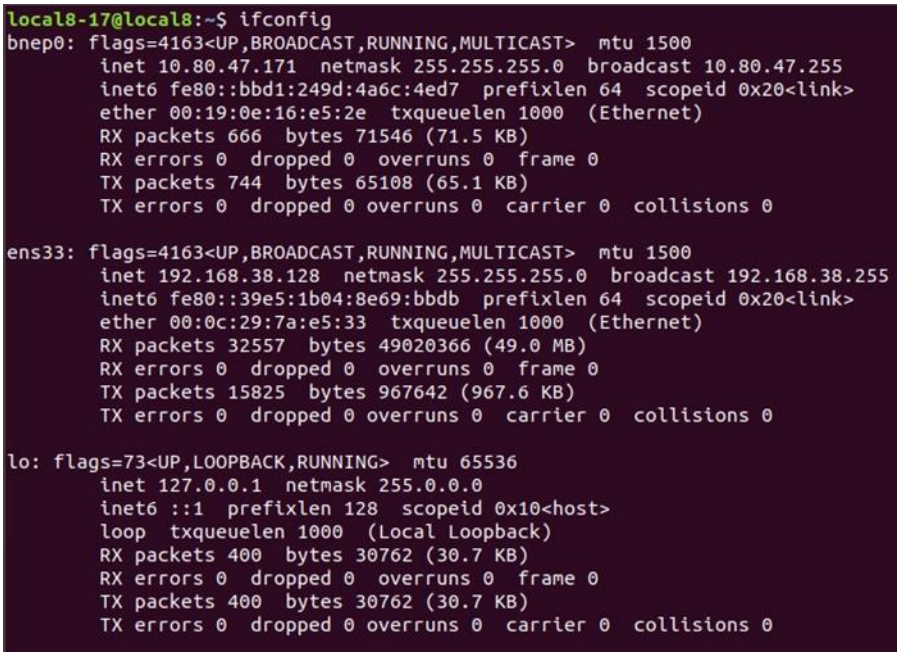

Figura 41. Salida del comando ifconfig en el PANU (local8)

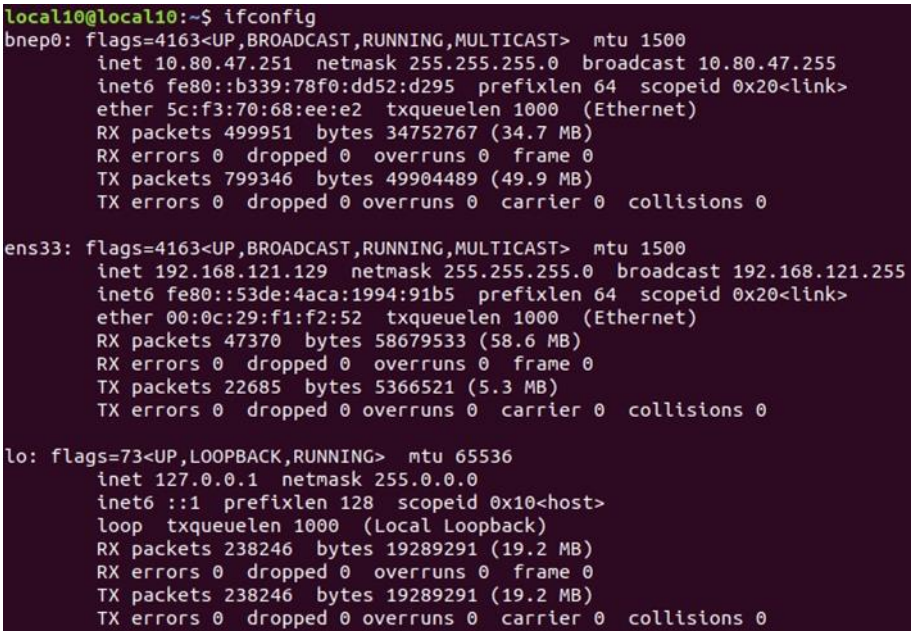

Figura 42. Salida del comando ifconfig en el PANU (local10)

A continuación se instalará “traceroute” que es una consola de diagnóstico que permite conocer la ruta que siguen los datagramas IP desde un host (punto de red) a un destino. Se obtiene además una estimación de la distancia a la que están los extremos de la comunicación. Para ello se ejecuta en el terminal el comando ***sudo apt install traceroute***

Ahora desde un PANU se muestra la ruta de los paquetes desde él hasta una página de Internet como por ejemplo Google.

Para ello se ejecuta el comando en el terminal desde un PANU ***traceroute www.google.com***. **Para acceder al exterior a través del local7 se debe desactivar la red Ethernet cableada del laboratorio ya que sino por defecto se accede por ahí.**

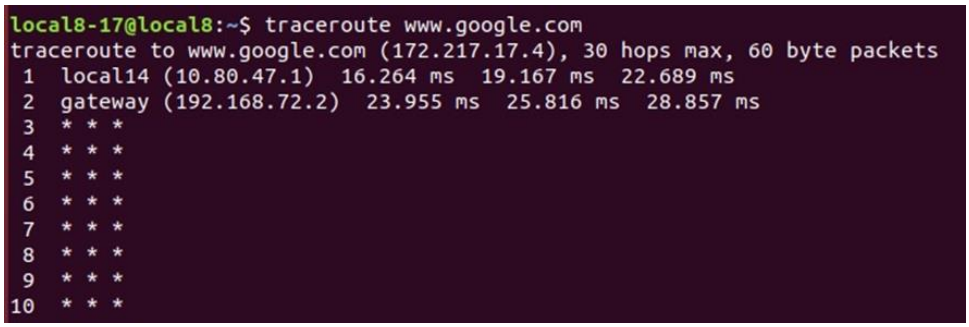

Figura 43. Traceroute desde local8 a Google

Se aprecia que el primer salto es a local7 a través de la IP privada 10.80.47.1. A través del Gateway de local7 que es 192.168.72.2 sale al exterior de nuestra red privada. Después iría al NAT del laboratorio que es Atlas y tiene como IP 192.168.110.11, luego al router del edificio y saldría desde la red de Unican pero no se puede apreciar desde las máquinas virtuales porque por defecto local7 actúa como NAT y al producirse una conexión entre dos NAT (entre local7 y Atlas) no se pueden ver todos los saltos posibles. Para poder ver la ruta completa, hay que cambiar la configuración de la máquina virtual desde el VMware, de modo NAT a modo bridge. Haciendo este cambio se aprecia que la máquina virtual pertenece a la subred del laboratorio de telemática en la interfaz ens33 cosa que antes no era así cuando se mostró con el ifconfig. En la imagen siguiente se ve de nuevo el ifconfig de la interfaz ens33 pero con la máquina virtual en modo bridge.

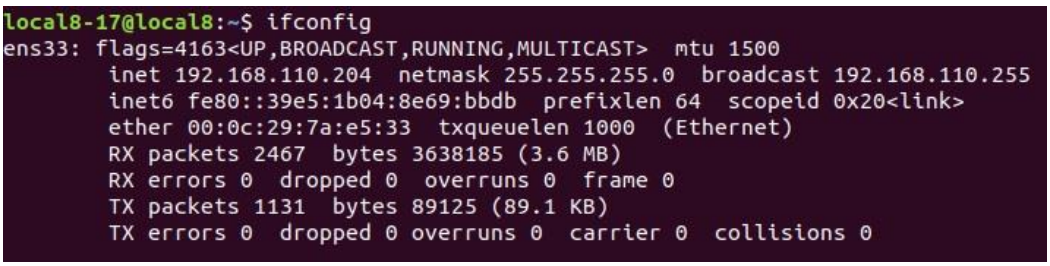

Figura 44. Ifconfig cambiando la configuración a modo bridge

Como se muestra, la dirección es 192.168.110.204 que pertenece a la subred del laboratorio de telemática. Una vez cambiado esto se vuelve a hacer el traceroute y se comprueba en la imagen siguiente que ahora sí que se ve la ruta completa de los paquetes.

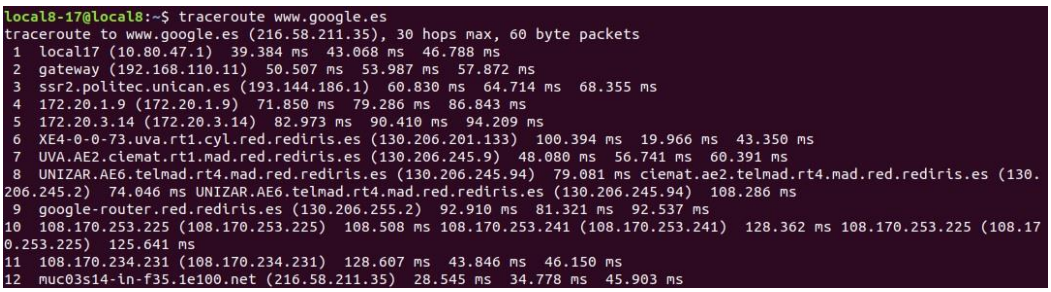

Figura 45. Traceroute desde local8 a Google en modo bridge

6.1. Configuración de los dispositivos

Como dispositivos Bluetooth, se dispone de dos tipos de “dongles”, uno es de la marca Belkin y el otro es del fabricante Targus **(Ver Hoja de características en el anexo).** Usan el mismo driver que es el BCM20702A0 y la versión de Bluetooth que utilizan es la 4.0. Se insertan los dongles en las ranuras USB de los ordenadores y cuando son reconocidos se observa el icono de Bluetooth en el menú superior de Ubuntu, lo que indica que ya se puede trabajar con Bluetooth.

Para ver las características principales de los dispositivos como es la dirección de la interfaz BD_ADDR, el tamaño en bytes de la MTU para los paquetes ACL y SCO, o el fabricante del chipset se ejecuta el comando ***hciconfig –a***. En la siguiente figura se pueden apreciar esas características.

```
local7-17@local7:~$ hciconfig -a
hci0:   Type: Primary  Bus: USB
        BD Address: 5C:F3:70:68:EE:E6  ACL MTU: 1021:8  SCO MTU: 64:1
        UP RUNNING PSCAN ISCAN
        RX bytes:29982 acl:272 sco:0 events:404 errors:0
        TX bytes:61638 acl:229 sco:0 commands:247 errors:0
        Features: 0xbf 0xfe 0xcf 0xfe 0xdb 0xff 0x7b 0x87
        Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
        Link policy: RSWITCH SNIFF
        Link mode: SLAVE ACCEPT
        Name: 'local7'
        Class: 0x1e0000
        Service Classes: Networking, Rendering, Capturing, Object Transfer
        Device Class: Miscellaneous,
        HCI Version: 4.0 (0x6)  Revision: 0x15ca
        LMP Version: 4.0 (0x6)  Subversion: 0x220e
        Manufacturer: Broadcom Corporation (15)
```

Figura 46. Características del dongle mediante el comando hciconfig -a

Ahora, el objetivo es conectarse entre las dos máquinas virtuales mediante Bluetooth. Para ello se dispone de dos opciones. La primera es desde el terminal usando diversos comandos y manejando el interfaz de comandos “bluetoothctl”. La segunda es desde la interfaz gráfica Blueman.

6.1.1. Bluetoothctl

Usando el controlador de Bluetooth “bluetoothctl” desde el terminal, se conectan dos dispositivos. Para ello se ejecutan los siguientes comandos en los dos ordenadores:

Paso 1: ***bluetoothctl***
Paso 2: ***discoverable on***. Permite que sea visible para otros dispositivos.
Paso 3: ***pairable on***. Permite que se empareje con otros dispositivos.
Paso 4: ***scan on***. Se ven todos los dispositivos Bluetooth que están al alcance y su dirección BD_ADDR.
Paso 5: ***pair BD_ADDR***. Empareja tu dispositivo al de la BD_ADDR que hayas puesto.
Paso 6: ***trust BD_ADDR***. Autoriza los dispositivos. Hay que autorizar en ambos dispositivos.
Paso 7: ***connect BD_ADDR***. Conecta los dispositivos. Es suficiente con ejecutarlo en un ordenador.

6.1.2. Blueman

Mediante la interfaz Blueman, se podrán conectar los dispositivos de una forma bastante sencilla e intuitiva. Lo primero es pinchar en el icono de Bluetooth y se abrirá un menú, en el que hay que seleccionar la opción “Devices”, como se muestra en la figura siguiente.

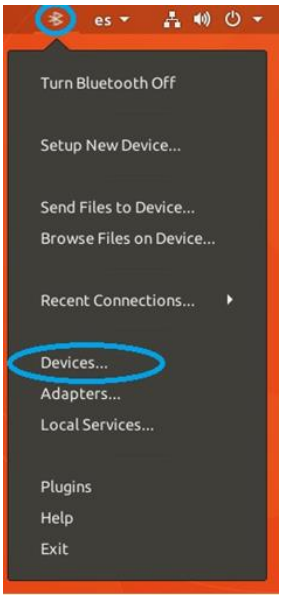

Figura 47. Menú principal de Blueman

Una vez dentro del menú “Bluetooth Devices”, se elige “Adapter”, después en “Preferences” y se selecciona la opción “always visible” en todos los ordenadores para que puedan ser descubiertos dentro del radio disponible. A continuación, se pincha en “Search”, y empezará la búsqueda de dispositivos visibles en el rango de búsqueda. Una vez acabada la búsqueda, se selecciona el dispositivo al que se quiere conectar, en este caso se quiere conectar con local7, y se procede a hacer el emparejado o “pairing” dando al icono de las llaves y la autorización o “trust” pinchando la estrella como se indica con círculos rojos en la imagen siguiente.

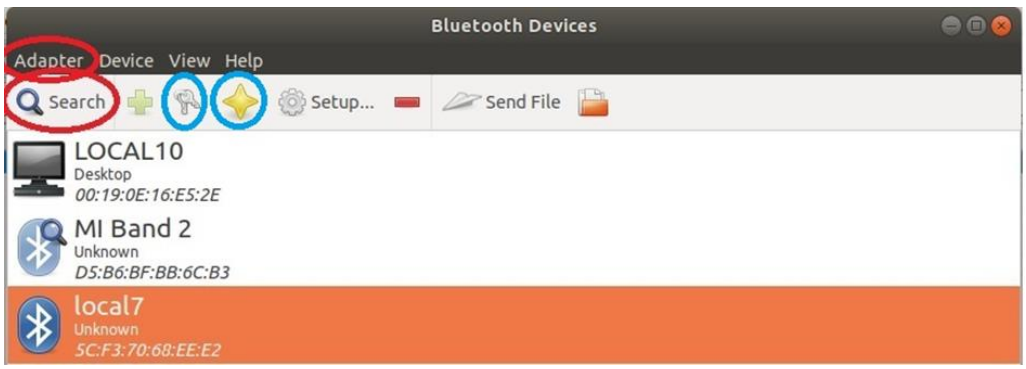

Figura 48. Menú Bluetooth Devices de Blueman

Una vez estén emparejados y autorizados saldrán los iconos de una llave y una estrella al lado del símbolo de Bluetooth del dispositivo, como vemos en la figura 49.

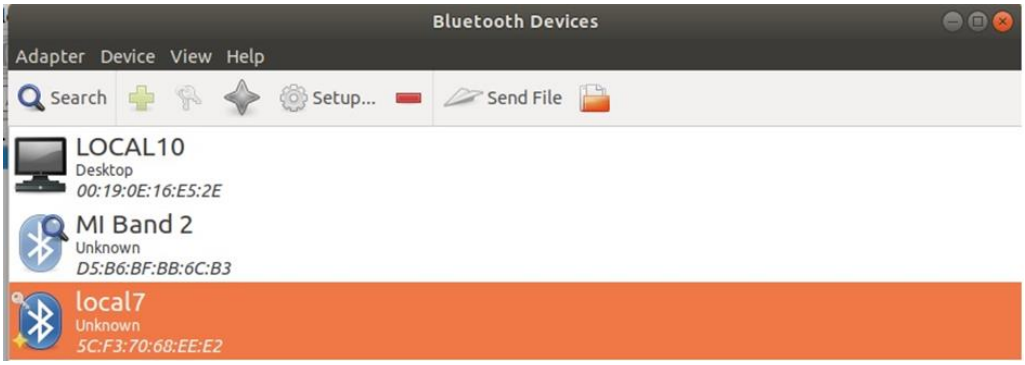

Figura 49. Emparejado y autorización en Blueman

Se tienen emparejados y autorizados los dispositivos, por lo que se procede a la conexión. Para ello se pincha en el dispositivo al que se quiere conectar, se presiona el botón derecho del ratón y se selecciona “Network Access Point” que permitirá conectarse al dispositivo que actuará como NAP. **Para que aparezca esta opción, se debe configurar local7 como NAP primero, ya que por defecto ningún dispositivo tiene esa función. Esto se explicará un poco más adelante.**

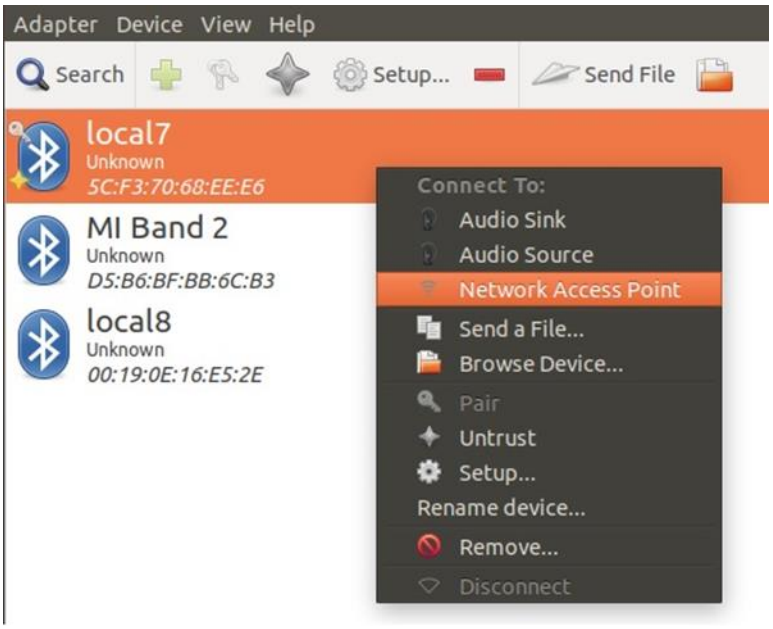

Figura 50. Conexión al NAP desde Blueman

Para comprobar que los dispositivos se encuentran conectados correctamente, existen varias opciones para verlo. En el menú de Blueman aparecen varios iconos de conexión, redondeados en rojo en la imagen que viene a continuación. En el menú superior el icono Bluetooth de Blueman se ha cambiado a color azul. Y por último, si se pincha en la flecha de la derecha del menú superior, se observa el número de dispositivos a los que se está conectado “1 Connected” y también que local7 actúa como NAP porque aparece “local7 Connected”. Todo esto se muestra en la figura 51.

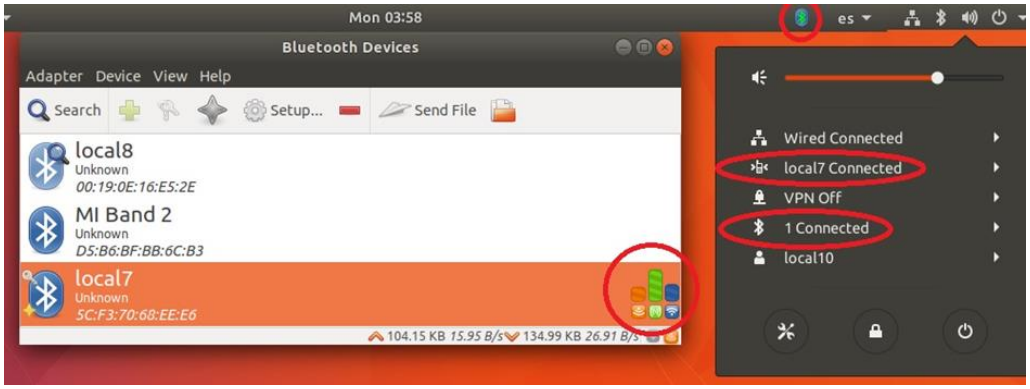

Figura 51. Dispositivo conectado en Blueman

6.1.3. Dispositivo que actúa como NAP

Como se ha comentado anteriormente, se necesita configurar un dispositivo como NAP antes de poder conectarnos a él y que este haga las funciones de un NAP. En nuestro caso el dispositivo que hace como NAP es local7, al que se le hacen los siguientes cambios.

Para empezar se selecciona el icono de Blueman del menú superior y se pincha en la pestaña “Local Services”.

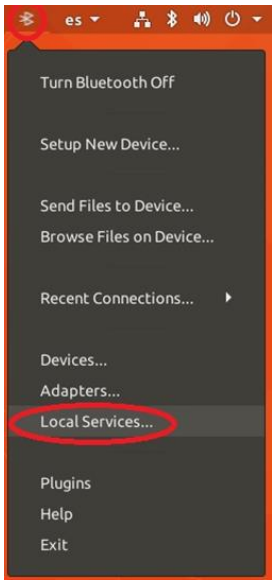

Figura 52. Local Services en el menú principal de Blueman

Dentro del menú “Local Services”, se selecciona en “Services” la opción Network Access Point (NAP). En la configuración del NAP, en DHCP server type se selecciona “dnsmasq” y como IP Address se deja la que sale por defecto, 10.80.47.1.

En PAN Support, se selecciona la opción “NetworkManager” y en DUN Support la opción “Blueman”. Debe quedar como en la figura 53. Con esta configuración, local7 ya puede actuar como NAP.

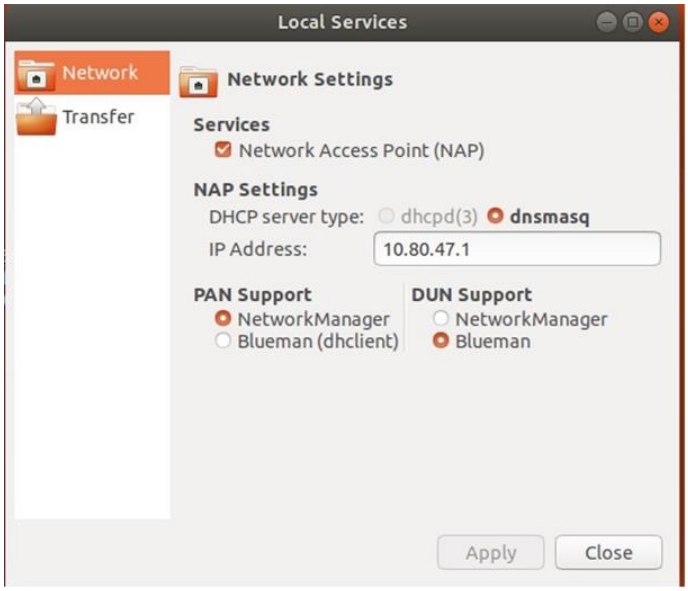

Figura 53. Configuración del NAP en Blueman

A la hora de configurar los PANUs, no es necesario realizar ningún cambio y se muestra cómo quedaría en la figura 54.

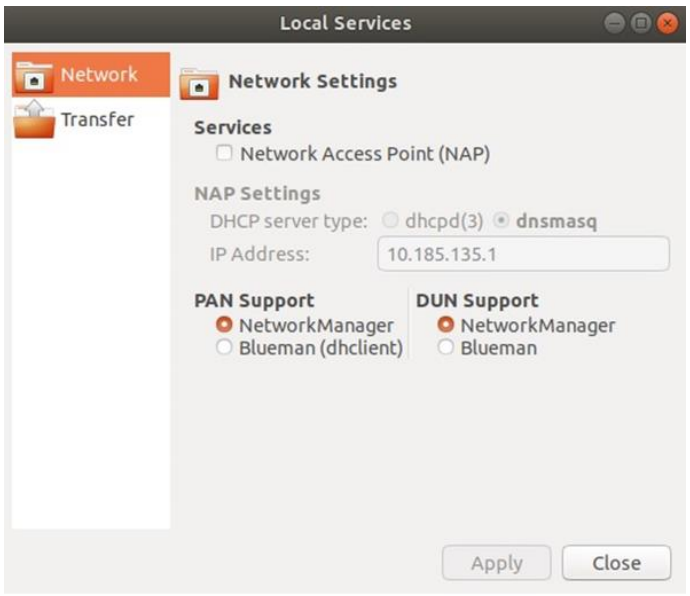

Figura 54. Configuración de los PANUs en Blueman

Para comprobar que el ordenador actúa como NAP, se van a ver los servicios y perfiles que ofrece. Para ello se pone el comando ***sudo sdptool browse local*** pero sale un error de compatibilidad de versión que no nos muestra esos servicios. Se ve en la figura 55.

```
local10@local10:~$ sudo sdptool browse local
[sudo] password for local10:
Failed to connect to SDP server on FF:FF:FF:00:00:00: No such file or directory
```

Figura 55. Error de compatibilidad del comando sdptool

Para conseguir que funcione este comando [19] , se modifica un archivo mediante el comando ***sudo nano /etc/systemd/system/dbus-org.bluez.service***. En la línea que pone “ExecStart =/usr/lib/bluetooth/bluetoothd” se añade “--compat” para arreglar los problemas de

compatibilidad de la versión 5 de BlueZ. La línea quedaría así: “ExecStart =/usr/lib/bluetooth/bluetoothd --compat”.

Se reinicia el demonio Bluetooth mediante los comandos ***sudo systemctl daemon-reload*** y ***sudo systemctl restart bluetooth***. Se cambian los permisos con ***sudo chmod 777 /var/run/sdp*** y ya funcionaría el comando sdptool.

Se vuelve a ejecutar el comando ***sudo sdptool browse local*** y se comprueban los servicios que soporta el NAP. En la figura siguiente se ve el servicio de red en el que actúa como NAP y se usarán los protocolos L2CAP y BNEP que viene indicado en el PSM (Protocol and Service Multiplexers) con el indicador 0x000f (15 en decimal).

Figura 56. Servicio de red NAP a través del comando sdptool browse local en local7

6.2. Monitorización del establecimiento de la conexión

Aquí se va a analizar el proceso de cómo se establece la conexión entre los PANUs con el NAP.

Para conectarse los dispositivos, como se he explicado anteriormente, existen dos formas de hacerlo ya sea con los comandos de “bluetoothctl” o con el interfaz gráfico Blueman. En cualquiera de los casos el que inicia la conexión siempre es el PANU queriéndose conectar al NAP. Aquí se va a analizar este proceso. Las siguientes capturas tomadas con el analizador de protocolos Wireshark están capturadas desde un PANU, concretamente desde local10, por lo que donde aparece “localhost” es local10. **Para ello hay que poner el Wireshark a capturar antes de que se establezca la conexión entre los dispositivos porque de lo contrario no se reconoce el protocolo BNEP como tal.**

En primer lugar, se analiza la trama L2CAP de Connection Request que manda local10 a nivel L2CAP indicando que se solicita establecer una conexión para el protocolo BNEP que viene indicado en el PSM (Protocol and Service Multiplexers) con el indicador 0x000f.

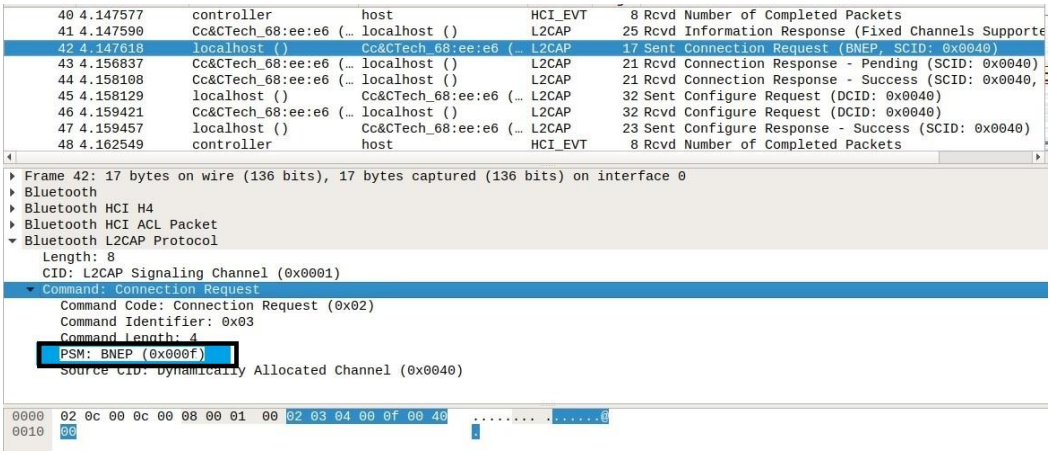

Figura 57. Connection Request para establecer conexión L2CAP para el transporte del protocolo BNEP

Ahora viene la réplica del NAP mandando dos tramas de Connection Response, una en la que el establecimiento de la conexión está pendiente y otra en la que manda la autorización de la conexión al PANU. Se observa que el identificador asignado a la conexión L2CAP es el CID 0x0040.

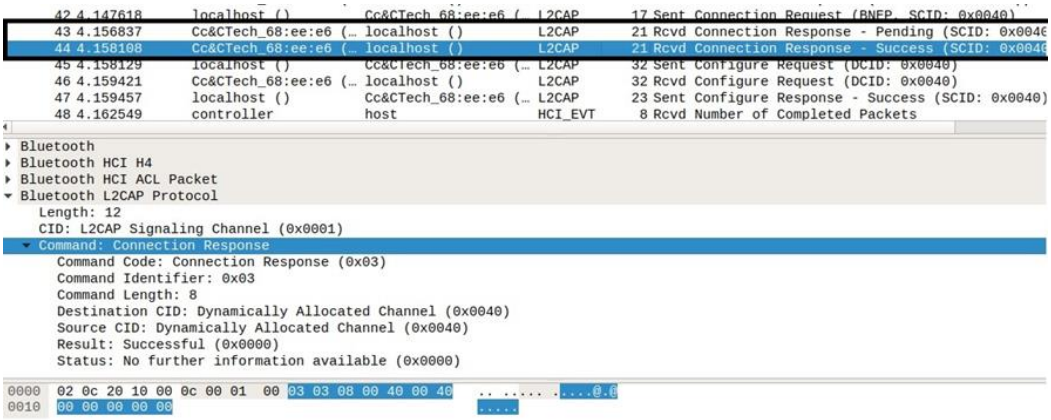

Figura 58. Connection Response para confirmación de la conexión L2CAP para el transporte del protocolo BNEP

Una vez que la conexión se ha establecido con éxito, se envía la trama L2CAP de Configure Request desde el PANU para establecer los parámetros de configuración acerca de la MTU máxima, retransmisión y control de flujo en el canal BNEP. Se observa en las figuras siguientes dicho intercambio de tramas.

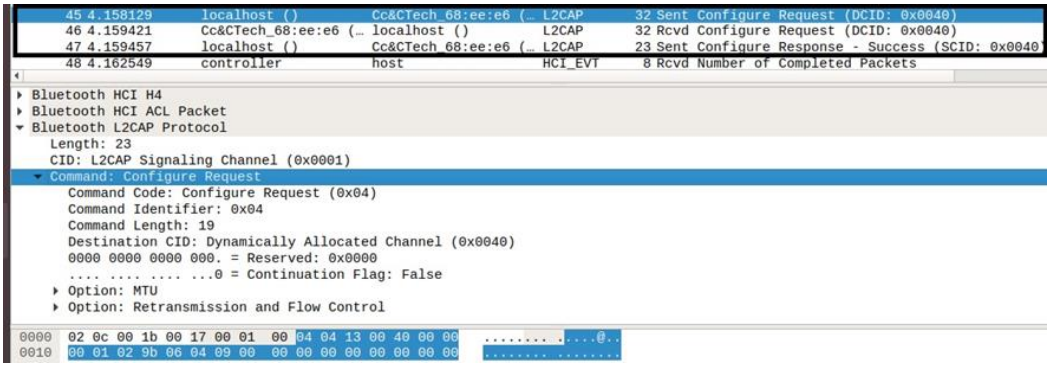

Figura 59. Configuration Request a nivel L2CAPpara establecer MTU, retransmisión y control de flujo

Se abren las opciones de MTU y la retransmisión y control de flujo. Se puede apreciar que la MTU máxima es de 1691 bytes y que el modo de retransmisión y control de flujo es el modo básico.

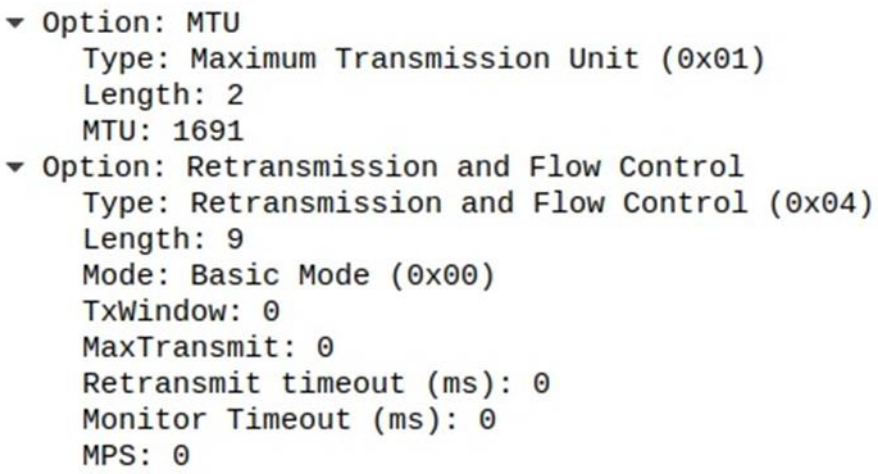

Figura 60. MTU máxima y modo de retransmisión y control de flujo

Después se confirman estas opciones con la trama Configuration Response que se mandan ambos dispositivos. Ahora ya se tiene establecida la conexión L2CAP para BNEP por lo que el protocolo ya podrá ser usado. La primera trama que se observa del protocolo BNEP es un Setup Connection Request que se manda desde el PANU. En la figura 61 se ve que es una trama de control y en el que se ven claramente las funciones de cada dispositivo a nivel PAN, la fuente (en este caso local10) es un PANU y el destino (local7) tiene la función de NAP.

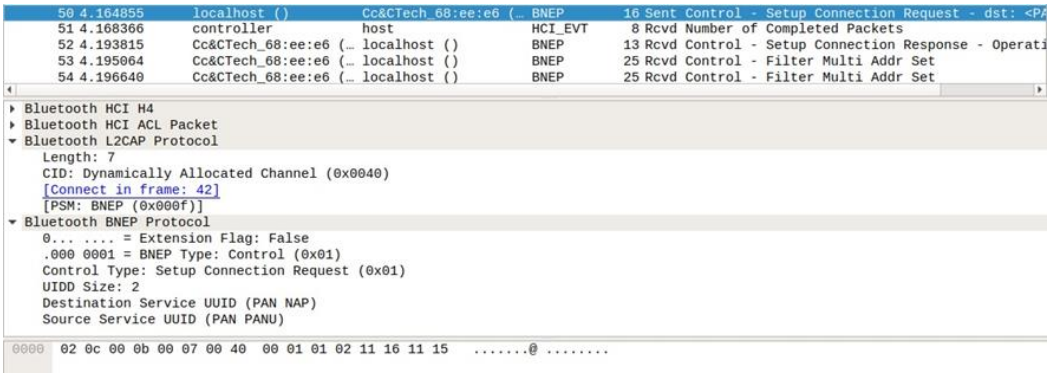

Figura 61. Paquete BNEP de configuración de la conexión Setup Connection Request

Una vez que el PANU envía el paquete Setup Connection Request, se espera la respuesta del NAP que acepta la conexión y por lo que ya se observa que los dispositivos configurados y conectados correctamente para utilizar BNEP a través de L2CAP.

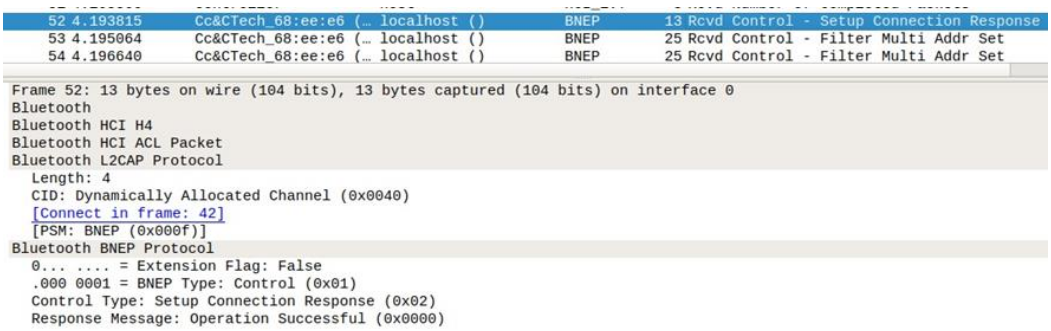

Figura 62. Setup Connection Response Successful

Un error bastante común que aparecía durante este proceso de conexión a nivel L2CAP fue el de Security Block. Si a la hora de conectarse al NAP desde un PANU, se ve que al cabo de uno o dos segundos la conexión se cancela es precisamente un problema de autorización entre los dispositivos. Por ello, es fundamental que además de estar emparejados, los dispositivos estén autorizados entre sí para utilizar el servicio correspondiente, en este caso, el protocolo BNEP, referido en la petición de conexión L2CAP con el PSM correspondiente (0x000f).

Se ve con Wireshark, en la figura 63, realizada desde el PANU. En este ejemplo, el PANU intenta conectarse al NAP, pero surge el problema de la desconexión. Por ello se captura con Wireshark y se ve como el dispositivo que actúa como NAP, en este caso con dirección 5C:F3:70:68:EE:E6, rechaza la conexión a nivel L2CAP con un mensaje de “Refused- security block”. En este caso el problema es que el NAP no le ha autorizado para poderse conectar a él. Este es uno de los niveles de seguridad de Bluetooth, como se explicó en el apartado 4.4.2.1 de teoría.

Es un error muy frecuente a la hora de trabajar con el controlador de Bluetooth (“bluetoothctl”) y que por ello merece la pena aclarar.

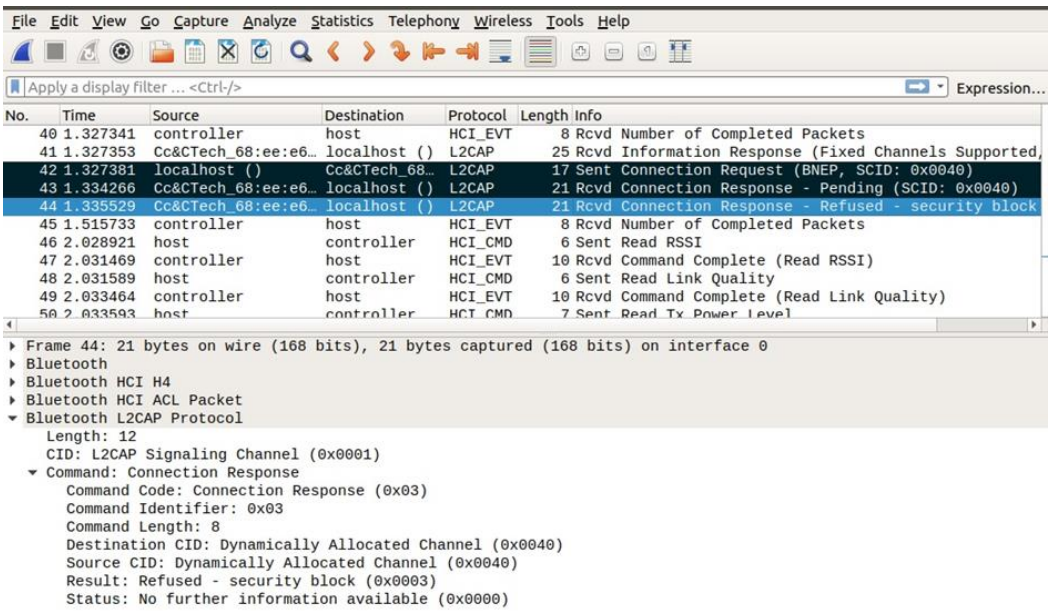

Figura 63. Error Security Block monitorizado con Wireshark

6.3. Medidas de rendimiento

Una vez creada la red PAN, se van a realizar una serie de medidas para caracterizar el rendimiento del escenario PAN basado en NAP (tethering). Para ello se instalarán los paquetes "nttcp" y "iperf" que son dos tipos de herramientas generadoras de patrones de tráfico, tanto TCP como UDP, para medir el rendimiento que ofrecen las interfaces de red.

Se instalan ejecutando los comandos ***sudo apt install nttcp*** y ***sudo apt install iperf*** Se comprueba que se han instalado correctamente con los comandos ***sudo dpkg –l | grep nttcp*** y ***sudo dpkg –l | grep iperf*** respectivamente.

Figura 64. Comprobación de instalación de los paquetes nttcp e iperf

Una vez instalados los generadores de tráfico, se va a medir el rendimiento entre PANU-NAP y entre PANU-PANU. Existe una diferencia significativa que es que en la conexión PANU-NAP solo hay un salto L2CAP mientras que en la conexión PANU-PANU hay dos saltos L2CAP, uno desde el PANU al NAP y otro desde el NAP al otro PANU. Esto influirá claramente en el throughput resultante como se verá a continuación.

6.3.1. Experimentos con Nttcp

En primer lugar se utilizará la herramienta nttcp. Para ello se pondrá un dispositivo como receptor y otro dispositivo como transmisor de tráfico. Este es un ejemplo ilustrativo de los comandos que se ejecutan en cada dispositivo y lo que aparecería en el terminal:

Primero se configura el receptor para escuchar mediante el comando ***nttcp –r –u –i***. La opción ***r*** establece que será el receptor (servidor); la ***u*** que se usará tráfico UDP; y la ***i*** lo ejecuta como demonio inet para poder ejecutar el cliente varias veces sin tener que volver a ejecutar el servidor cada vez que se vaya a transmitir algo.

Una vez esté el receptor preparado, se ejecuta el siguiente comando en el transmisor: ***nttcp -t -u -l LONG -n N_FRAG -T DIR_SERV***. La opción ***t*** establece que será el transmisor (cliente); la ***u*** que se usará tráfico UDP; la opción ***l*** permite especificar la longitud en bytes del campo de datos de cada datagrama UDP (luego se explicará por qué se ponen diferentes bytes de datos (***LONG***) en cada caso); ***n*** dice el número de paquetes que se envían (***N_FRAG***); se pone ***T*** para que salga la cabecera en la información por pantalla; y, por último, la dirección del equipo que se ha configurado como servidor (***DIR_SERV***).

En la siguiente figura, se muestra un ejemplo de los resultados de la transmisión.

```
local8-17@local8:~$ nttcp -t -u -l 986 -n 1000 -T 10.80.47.1
     Bytes  Real s   CPU s Real-MBit/s  CPU-MBit/s   Calls  Real-C/s   CPU-C/s
l   986000     3.75    0.00      2.1033   1590.0020    1003    267.45  202177.0
1   986000     4.58    0.01      1.7241    605.0936    1001    218.79   76787.4
```

Figura 65. Ejemplo de transmisión mediante la herramienta nttcp

La primera línea indica los resultados medidos por el transmisor y la segunda los observados por el receptor. Lo que realmente interesa es la columna de Real-Mbit/s que la del throughput. Simplemente es la división de los Bytes que se transfieren entre el tiempo real que se tarda.

Ahora se repiten las medidas. **En primer lugar se enviarán 1000 datagramas UDP ajustando el tamaño de datos al del paquete 3-DH5 Bluetooth**. La selección de los bytes de datos se hace para maximizar el rendimiento minimizando el "overhead" introducido por las cabeceras. Para llegar a la cifra correcta, se parte del tamaño del paquete 3-DH5 (1021 Bytes) que es el que mayor rendimiento y velocidad proporciona (como se ve en la tabla número 1). A partir de ahí es necesario revisar la sobrecarga de los paquetes en las diferentes capas del "stack", tal y como se muestra en la figura siguiente.

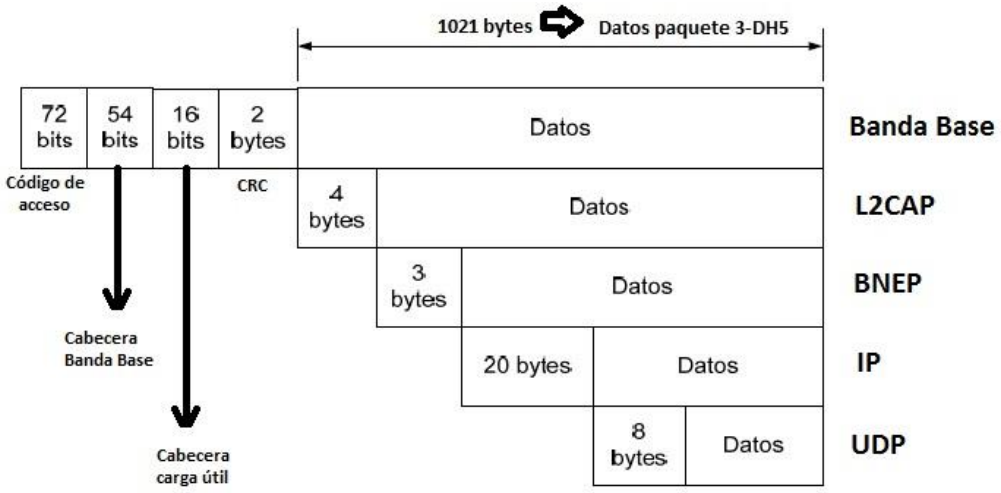

Figura 66. Estructura de los datagramas UDP sobre paquetes ACL 3-DH5

El tamaño de la cabecera de la capa BNEP depende del tipo de paquete BNEP. Para saber qué tipo de paquete BNEP se usa, se recurre a Wireshark poniéndolo a capturar durante una transmisión de prueba. Como se comentó anteriormente, se va a medir el rendimiento entre PANU-NAP y entre PANU-PANU; por lo que haremos dos transmisiones de prueba, una de cada tipo.

La primera prueba es entre PANU-NAP. Se envía tráfico desde local10 hasta el NAP que es local7. Se captura con Wireshark y se selecciona el protocolo BNEP. Ahí aparece que el tipo es BNEP_COMPRESSED_ETHERNET (0x02) y se ven los 2 bytes añadidos que indican el tipo de protocolo de red (0x0800) que en este caso es IPv4 siendo en total 3 bytes la encapsulación BNEP. **Merece la pena reseñar que el analizador Wireshark tiene que capturar la conexión al NAP y luego la transmisión de datos sin detener el analizador porque de lo contrario no reconocerá la cabecera BNEP como tal dado que no guarda la información del identificador L2CAP asociado a esa conexión, por lo que no aplicará el diseptor BNEP a los paquetes capturados. Esto se comenta también en el apartado de cómo se conectan los dispositivos.**

También se aprecia la dirección IP de la fuente 10.80.47.251 que es local10 y la IP del destino que es 10.80.47.1 que es local7. En la figura 67 se observa una captura de Wireshark de la conexión PANU-NAP.

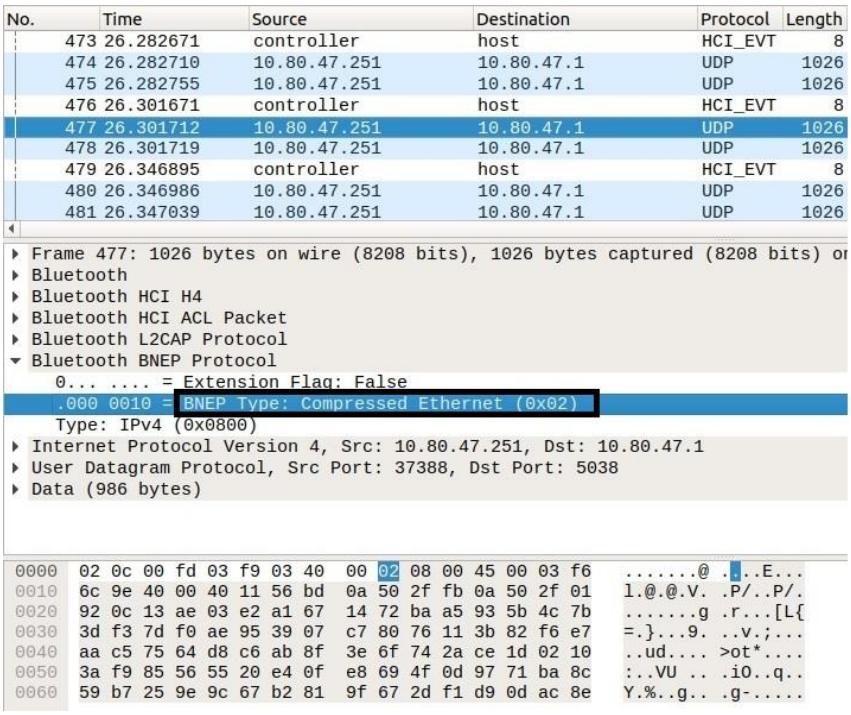

Figura 67. Tipo de paquete BNEP en la conexión PANU-NAP

La segunda prueba es entre PANU-PANU. Se envía tráfico desde local10 hasta el otro PANU que es local8. Se captura con Wireshark y se pincha en el protocolo BNEP. Ahí aparece que el tipo es BNEP_COMPRESSED_ETHERNET_DEST_ONLY (0x04) y se ven los 8 bytes añadidos. 6 bytes indican la dirección destino (00:19:0e:16:e5:2e) y los otros 2 bytes indican el tipo de protocolo de red (0x0800) que en este caso es IPv4 siendo en total 9 bytes la encapsulación BNEP. La dirección destino coincide con la BD_ADDR de local8.

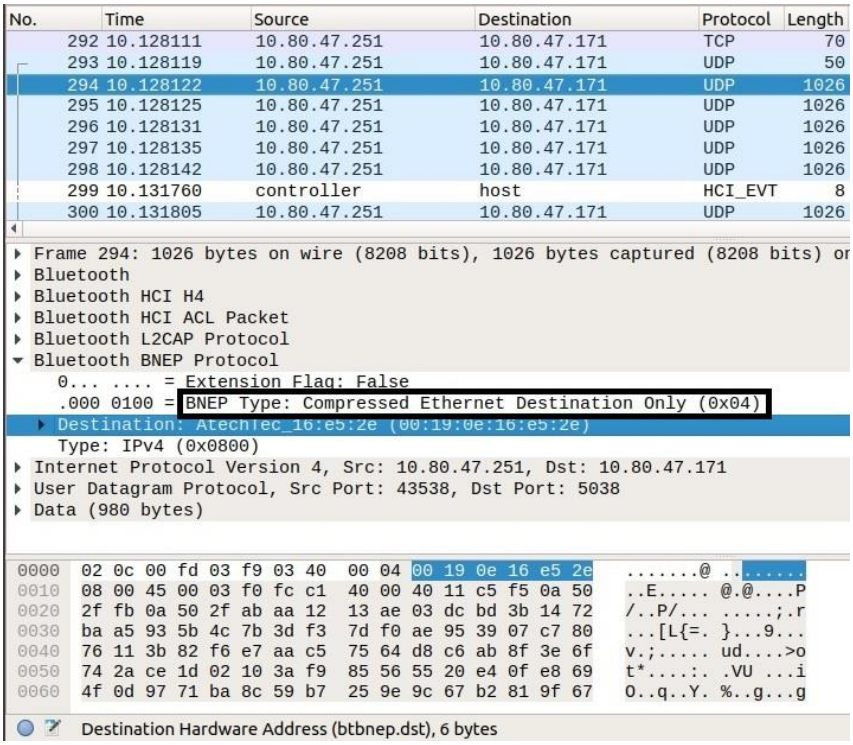

Figura 68. Tipo de paquete BNEP en la conexión PANU-PANU

Después de comprobar que hay distintos tipos de paquetes BNEP tendremos que ajustar los bytes de datos para evitar que los paquetes UDP se dividan al llegar a banda base y así disminuir el "overhead" introducido por las cabeceras.

Para la conexión PANU-NAP habrá el siguiente número de bytes de datos:

1021 paquete 3-DH5 - (4 - 3 - 20 - 8) cabeceras = 986 Bytes

4 de cabecera L2CAP, 3 de cabecera BNEP, 20 de cabecera IP y 8 de cabecera UDP.

Para la conexión PANU-PANU serán:

1021 paquete 3-DH5 - (4 - 9 - 20 - 8) cabeceras = 980 Bytes

4 de cabecera L2CAP, 9 de cabecera BNEP, 20 de cabecera IP y 8 de cabecera UDP.

Una vez se sabe el número de bytes que se van a enviar ya tenemos todos los datos para empezar con las medidas, pero antes se van a realizar unos cálculos teóricos para compararlos con los resultados posteriores.

Se procede a hacer un cálculo teórico aproximado de las velocidades de transmisión esperadas. El tiempo necesario para enviar un paquete 3-DH5 es de 6 timeslots; 5 slots es el tiempo que tarda en transmitir el paquete y uno más porque sólo comienza a enviar cada paquete en los slots pares, o en los impares si el que transmite es el esclavo en el que viaja la confirmación. En definitiva:

T_3DH5 = 6 x timeslot = 6 x 625µs = 3,75 ms

Por lo tanto, el rendimiento máximo que se obtiene es:

$$\mathbf{V\_real} = \frac{\mathbf{986\ bytes\ x\ 8}}{T\_3DH5} = \mathbf{2{,}103\ Mbits/s}$$

$$\mathbf{V\_real2} = \frac{\mathbf{980\ bytes\ x\ 8}}{T\_3DH5} = \mathbf{2{,}090\ Mbits/s}$$

Mientras que la velocidad nominal o velocidad en la capa banda base que establece la definición de Bluetooth para la transmisión con paquetes 3-DH5 viene determinada por el siguiente cálculo:

$$\mathbf{V\_nominal} = \frac{\mathbf{1021\ bytes\ x\ 8}}{T\_3DH5} = \mathbf{2{,}178\ Mbits/s}$$

Volviendo a la figura 67, se observa cómo los resultados medidos en el transmisor coinciden exactamente con los mencionados para la conexión PANU-NAP.

A continuación, se empezará con el envío de tráfico UDP:

Conexión PANU-NAP

La primera medida se hace para una conexión de un único salto Bluetooth. Se pone el NAP (local7) como receptor y el PANU (local10) como transmisor y ponemos los siguientes comandos:

Receptor: ***nttcp -r -u –i***
Transmisor: ***nttcp -t -u -l 986 -n 1000 -T 10.80.47.1***

Se van a coger 7 muestras de los Throughput, de las cuáles descartaremos la más alta y la más baja y se hará la media de las restantes.

Transmisión (Mb/s)	Recepción (Mb/s)
2,1033	1,7241
~~2,0404~~	~~1,6761~~
~~2,1109~~	1,7232
2,0936	1,6926
2,0946	1,6868
2,0957	~~1,7341~~
2,0548	1,7092

Tabla 4. Throughput PANU-NAP con nttcp

Media transmisión: 2,088 Mbits/s
Media recepción: 1,707 Mbits/s

Se obtiene una velocidad media de 1,707 Mb/s, casi un 20% inferior a la calculada teóricamente (2,1 Mb/s). Esta diferencia puede ser debida al retardo introducido por el procesado en el NAP (alrededor de 800 µs por datagrama UDP/IP).

Conexión PANU-PANU

La segunda medida se hace para una conexión de dos saltos Bluetooth, uno PANU-NAP y otro NAP-PANU. Se pone el PANU (local8) como receptor y el PANU (local10) como transmisor y ponemos los siguientes comandos:

Receptor: ***nttcp -r -u –i***
Transmisor: ***nttcp -t -u -l 980 -n 1000 -T 10.80.47.171***

Se van a coger 7 muestras de los Throughput, de las cuáles descartaremos la más alta y la más baja y se hará la media de las restantes.

Transmisión (Mb/s)	Recepción (Mb/s)
~~1,1200~~	0,8493
1,1976	~~0,8722~~
~~1,2026~~	0,8529
1,1677	0,8389
1,1929	~~0,7725~~
1,1815	0,8094
1,1708	0,8051

Tabla 5. Throughput PANU-PANU con nttcp

Media transmisión: 1,182 Mbits/s
Media recepción: 0,831 Mbits/s

Se obtiene una velocidad media de 0,831 Mbits/s; es decir que se reduce alrededor del 50% la velocidad que se lograba con un solo salto (1,707 Mbits/s).

Dado que en muchas ocasiones las aplicaciones utilizan como tamaño máximo el que se maneja en las redes cableadas, se va a repetir el proceso pero enviando paquetes IP de 1500 bytes ajustados a la MTU de Ethernet. En este caso los paquetes serán segmentados en el nivel banda base en 2 paquetes 3-DH5. Para entender mejor el proceso de segmentación y reensamblado en el nivel banda base, se adjunta la figura siguiente para tener una perspectiva más visual.

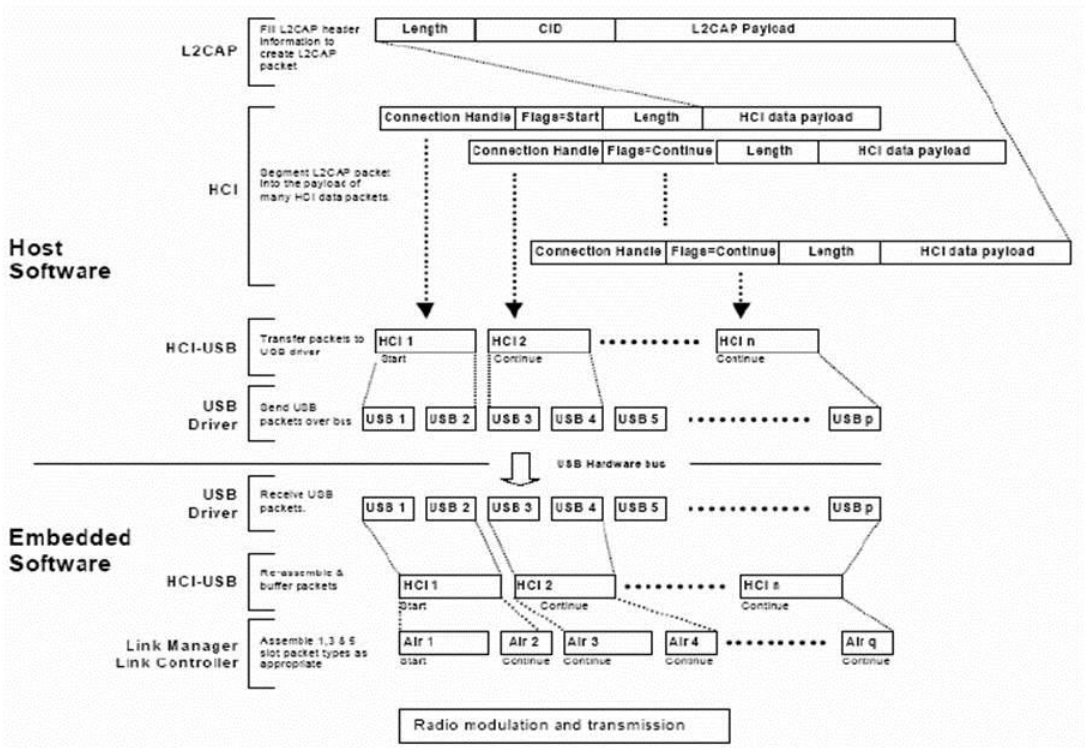

Figura 69. Segmentación y reensamblado en el nivel banda base

Posteriormente se muestran dos imágenes capturadas desde el analizador Wireshark, enviando este tipo de paquetes IP de 1500 bytes ajustados a la MTU de Ethernet.

No.	Time	Source	Destination	Protocol	Length	Info
36	0.19…	localhost ()	remote ()	L2CAP	79	Sent Connection oriented channel
37	0.20…	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
38	0.20…	remote ()	localhost ()	L2CAP	64	Rcvd Connection oriented channel
39	0.20…	remote ()	localhost ()	L2CAP	44	Rcvd Connection oriented channel
40	0.21…	remote ()	localhost ()	HCI_ACL	1026	Rcvd [Reassembled in #41]
41	0.21…	remote ()	localhost ()	L2CAP	491	Rcvd Connection oriented channel
42	0.21…	remote ()	localhost ()	HCI_ACL	1026	Rcvd [Reassembled in #43]
43	0.22…	remote ()	localhost ()	L2CAP	491	Rcvd Connection oriented channel
44	0.23…	remote ()	localhost ()	HCI_ACL	1026	Rcvd [Reassembled in #45]
45	0.23…	remote ()	localhost ()	L2CAP	491	Rcvd Connection oriented channel
46	0.24…	remote ()	localhost ()	HCI_ACL	1026	Rcvd [Reassembled in #47]
47	0.24…	remote ()	localhost ()	L2CAP	491	Rcvd Connection oriented channel
48	0.26…	remote ()	localhost ()	HCI_ACL	1026	Rcvd [Reassembled in #49]
49	0.27…	remote ()	localhost ()	L2CAP	491	Rcvd Connection oriented channel
50	0.28…	remote ()	localhost ()	HCI_ACL	1026	Rcvd [Reassembled in #51]
51	0.28…	remote ()	localhost ()	L2CAP	491	Rcvd Connection oriented channel
52	0.29…	remote ()	localhost ()	HCI_ACL	1026	Rcvd [Reassembled in #53]
53	0.29…	remote ()	localhost ()	L2CAP	491	Rcvd Connection oriented channel
54	0.31…	remote ()	localhost ()	HCI_ACL	1026	Rcvd [Reassembled in #55]

```
▸ Frame 40: 1026 bytes on wire (8208 bits), 1026 bytes captured (8208 bits) on interface 0
▸ Bluetooth
▾ Bluetooth HCI H4
    [Direction: Rcvd (0x01)]
    HCI Packet Type: ACL Data (0x02)
▾ Bluetooth HCI ACL Packet
    .... 0000 0000 1011 = Connection Handle: 0x00b
    ..10 .... .... .... = PB Flag: First Automatically Flushable Packet (2)
    00.. .... .... .... = BC Flag: Point-To-Point (0)
    Data Total Length: 1021
    [This PDU is reassembled in frame: 41]
```

Figura 70. Segmentación en el nivel banda base del paquete de 1500 bytes (Paquete 1)

En la primera imagen se muestra que la longitud total de datos ACL es de 1021 bytes. Aparecen también los campos de los paquetes HCI como Connection Handle, y las flags.

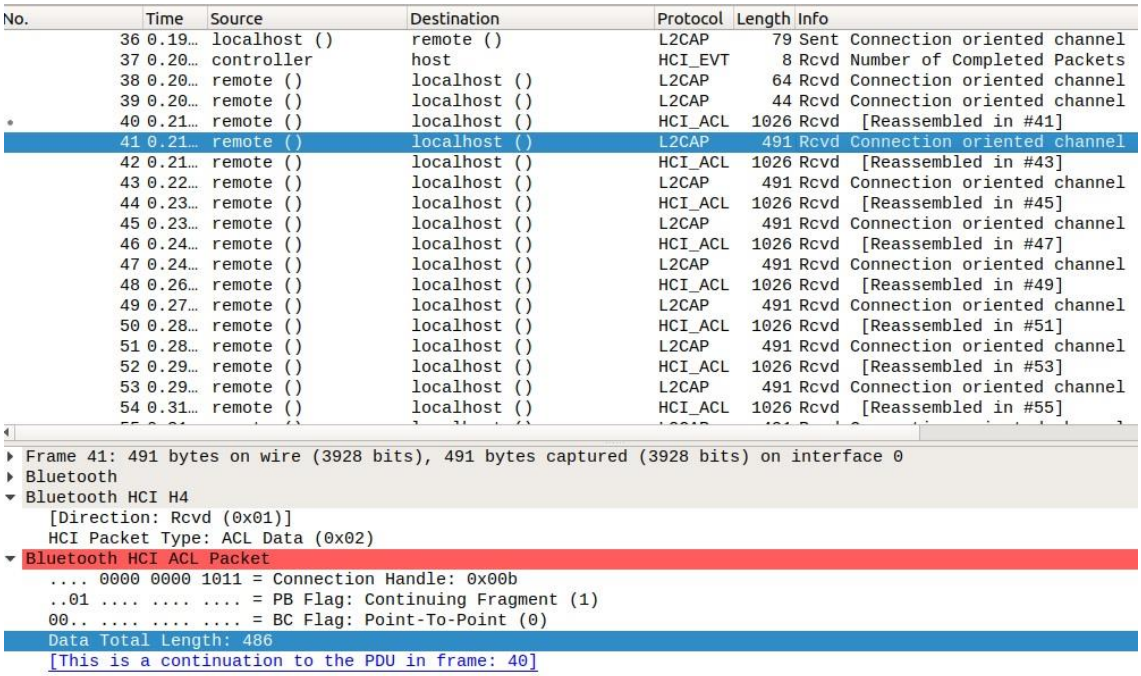

Figura 71. Segmentación en el nivel banda base del paquete de 1500 bytes (Paquete 2)

En esta imagen se aprecia que es la continuación del paquete anterior, como se ve en la PB flag y posteriormente lo indica el propio Wireshark en azul. La longitud total de datos ACL es de 486 bytes, por lo que transporta en torno al 50% de su capacidad.

Si sumamos los 1021 bytes del primer paquete con los 486 bytes del segundo son 1507 bytes. 1500 son bytes de datos, 4 de cabecera L2CAP y 3 del protocolo BNEP.

Antes de medir el throughput enviando paquetes IP de 1500 bytes ajustados a la MTU de Ethernet, se realiza un cálculo teórico para compararlo con los resultados posteriores.

T_3DH5 = 6 x timeslot = 6 x 625µs = 3,75 ms

$$\textbf{V\_real} = \frac{\textbf{1472 bytes x 8}}{2\ x\ T\_3DH5} = \textbf{1,57 Mbits/s}$$

La velocidad en la recepción no podrá ser en ningún caso mayor que el resultado obtenido.

A continuación, se empieza con el envío de tráfico UDP:

Conexión PANU-NAP

Se pone el NAP (local7) como receptor y el PANU (local10) como transmisor y ponemos los siguientes comandos:

Receptor: ***nttcp -r -u –i***
Transmisor: ***nttcp -t -u -l 1472 -n 1000 -T 10.80.47.1***

Se van a coger 7 muestras de los Throughput, de las cuáles descartaremos la más alta y la más baja y se hará la media de las restantes.

Transmisión (Mb/s)	Recepción (Mb/s)
1,7151	~~1,5639~~
~~1,4417~~	~~1,2762~~
1,6158	1,4676
1,7274	1,5639
1,6803	1,3351
~~1,7499~~	1,5323
1,6946	1,5306

Tabla 6. Throughput PANU-NAP con MTU de 1500 bytes con nttcp

Media transmisión: 1,686 Mbits/s
Media recepción: 1,485 Mbits/s

Se obtiene una velocidad de 1,485 Mbits/s que es algo menor que en el caso anterior (1,707 Mbits/s) debido a la segmentación de los paquetes.

Conexión PANU-PANU

Se pone el PANU (local8) como receptor y el PANU (local10) como transmisor ejecutando los siguientes comandos:

Receptor: ***nttcp -r -u –i***
Transmisor: ***nttcp -t -u -l 1472 -n 1000 -T 10.80.47.171***

Se van a coger 7 muestras de los Throughput, de las cuáles descartaremos la más alta y la más baja y se hará la media de las restantes.

Transmisión (Mb/s)	Recepción (Mb/s)
0,9924	0,6867
~~1,0223~~	0,6848
0,9939	~~0,6501~~
0,9979	~~0,6925~~
~~0,9795~~	0,6639
1,0164	0,6651
0,9964	0,6762

Tabla 7. Throughput PANU-PANU con MTU de 1500 bytes con nttcp

Media transmisión: 0,999 Mbits/s
Media recepción: 0,675 Mbits/s

Se obtiene una velocidad media de 0,675 Mbits/s; es decir que se reduce alrededor del 50% la velocidad que se lograba con un solo salto (1,485 Mbits/s) al igual que en el caso anterior en el que se ajustó el tamaño de datos al del paquete 3-DH5 ACL.

6.3.2. Experimentos con Iperf

Con la herramienta iperf se puede comprobar también el rendimiento entre PANU-NAP y entre PANU-PANU. Este es un ejemplo ilustrativo de los comandos que se ponen en cada dispositivo y lo que aparecerá en el terminal:

Primero se pone el receptor a escuchar mediante el comando ***iperf –s –u –l LONG*** La opción ***s*** establece que será el servidor; la ***u*** que se usará tráfico UDP; y la ***l*** la longitud en bytes del campo de datos de cada datagrama UDP que será ***LONG***.

Una vez se tenga el receptor preparado, se ejecuta el siguiente comando en el transmisor o cliente ***iperf –c DIR_IP_DEST –u –b ANCHO –l LONG –t TIEMPO –i INTERVALO***

La opción ***c*** establece que será el cliente; a continuación en ***DIR_IP_DEST*** se pone la dirección IP del servidor; la ***u*** que se usará tráfico UDP; la opción ***b*** indica el ancho de banda máximo al que podemos transmitir siendo este ***ANCHO*** (3m equivale a 3 Mbits/s de velocidad bruta máxima); la opción ***l*** permite especificar la longitud en bytes del campo de datos de cada datagrama UDP que será ***LONG***; la opción ***t*** permite especificar el tiempo en el que se transmiten los datagramas que será ***TIEMPO***; la opción ***i*** indica el intervalo en el que aparecen los resultados de la transmisión en el cliente que será ***INTERVALO***

En las figuras siguientes vemos los ejemplos de los resultados de poner el comando iperf tanto en el cliente como en el servidor. En el cliente, durante la transmisión, aparecen los bytes que se transmiten en el intervalo que se especifica en el comando.

```
local8-17@local8:~$ iperf -c 10.80.47.1 -u -b 3m -l 986 -t 15 -i 3
------------------------------------------------------------
Client connecting to 10.80.47.1, UDP port 5001
Sending 986 byte datagrams, IPG target: 2629.33 us (kalman adjust)
UDP buffer size:  208 KByte (default)
------------------------------------------------------------
[  3] local 10.80.47.171 port 56340 connected with 10.80.47.1 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0- 3.0 sec   558 KBytes  1.53 Mbits/sec
[  3]  3.0- 6.0 sec   647 KBytes  1.77 Mbits/sec
[  3]  6.0- 9.0 sec   601 KBytes  1.64 Mbits/sec
[  3]  9.0-12.0 sec   647 KBytes  1.77 Mbits/sec
[  3] 12.0-15.0 sec   601 KBytes  1.64 Mbits/sec
[  3]  0.0-15.0 sec  2.98 MBytes  1.67 Mbits/sec
[  3] Sent 3172 datagrams
[  3] Server Report:
[  3]  0.0-14.7 sec  2.98 MBytes  1.71 Mbits/sec   0.000 ms    0/ 3172 (0%)
```

Figura 72. Ejemplo de iperf en el cliente

```
local7-17@local7:~$ iperf -s -u -l 986
------------------------------------------------------------
Server listening on UDP port 5001
Receiving 986 byte datagrams
UDP buffer size:  208 KByte (default)
------------------------------------------------------------
[  3] local 10.80.47.1 port 5001 connected with 10.80.47.171 port 56340
[ ID] Interval       Transfer     Bandwidth        Jitter   Lost/Total Datagrams
[  3]  0.0-14.7 sec  2.98 MBytes  1.71 Mbits/sec  18.699 ms    0/ 3172 (0%)
```

Figura 73. Ejemplo de iperf en el servidor

A continuación, se empezará con la simulación de tráfico UDP mediante iperf:

Conexión PANU-NAP

Se realizan las medidas PANU-NAP poniendo los siguientes comandos:

Servidor: ***iperf –s –u –l 986***
Cliente: ***iperf –c 10.80.47.1 -u –b 3m –l 986 –t 15–i 3***

Los resultados de la transmisión son los siguientes:

Intervalo(s)	Transferido (MBytes)	Velocidad(Mb/s)	Jitter (ms)	Datagramas perdidos
0.0-14.7	2,98	1,71	18,699	0 de 3172

Tabla 8. Resultado PANU-NAP con iperf

Se comprueba que se han transmitido en 2.98 MBytes en 15 segundos con una velocidad media de 1.71 Mbits/s, exactamente la misma que salía en con el comando nttcp anteriormente para este caso (1,707 Mb/s). El total de datagramas es de 3172 de los cuales se han recibido todos sin pérdida.

El jitter es de 18.69 ms e indica la variación del retardo entre los datagramas recibidos. Se aprecia que es inferior a 100 ms.

Conexión PANU-PANU

En este caso se realiza entre PANU-PANU pasando por el NAP para ver las diferencias que se encuentran, siendo local10 el servidor y local8 el cliente. Los comandos que hay que poner son los siguientes:

Servidor: ***iperf –s –u –l 980***
Cliente: ***iperf –c 10.80.47.251 -u -b 3m –l 980 –t 15–i 3***

Los resultados de esta transmisión en el servidor son estos:

Intervalo(s)	Transferido (MBytes)	Velocidad(Mb/s)	Jitter (ms)	Datagramas perdidos
0.0-18.1	1,86	0,863	29,580	0 de 1992

Tabla 9. Resultado PANU-PANU con iperf

Sin embargo, en el cliente aparece que el intervalo es de 0 a 15.3 segundos y que la velocidad es de 1,02 Mbits/s. Se produce una pérdida de velocidad debido al paso por el NAP y las dos conexiones L2CAP que existen.

La velocidad es el 50% menos que en el caso PANU-NAP y por ello se transmiten menos datagramas también, en este son 1992 datagramas transmitidos. En el mismo tiempo se enviaron 3172 datagramas en la conexión PANU-NAP. También se aprecia que el jitter es algo mayor que en el primer caso.

Se va a repetir el proceso pero enviando paquetes de 1500 bytes como la MTU de Ethernet. En este caso los paquetes serán segmentados en el nivel L2CAP en 2 paquetes 3-DH5, aumentando así el "overhead" de banda base, pero se disminuye el de las capas superiores pues la cabecera de estas sólo viaja en el primer paquete y no en todos como ocurría en el caso anterior.

Conexión PANU-NAP

Se realizan las medidas PANU-NAP ejecutando los siguientes comandos:

Servidor: ***iperf –s –u –l 1472***
Cliente: ***iperf –c 10.80.47.1 -u –b 3m –l 1472 –t 15–i 3***

Los resultados de esta transmisión son los siguientes:

Intervalo(s)	Transferido (MBytes)	Velocidad(Mb/s)	Jitter (ms)	Datagramas perdidos
0.0-15.8	2,87	1,53	33,488	0 de 2046

Tabla 10. Resultado PANU-NAP con iperf con 1472 bytes de datos

Se comprueba que se han transmitido en 2.87 MBytes en 15 segundos con una velocidad media de 1.53 Mbits/s. Es algo inferior la velocidad debido a la segmentación de los paquetes que en el caso que iban ajustados los datos al paquete DH5 de Bluetooth que era de 1,71 Mbits/s.

Se observa el jitter (33,48 ms) y los datagramas que se han perdido (0 de 2046).

Conexión PANU-PANU

Por último se realizan las medidas PANU-PANU poniendo los siguientes comandos, siendo local10 el servidor y local8 el cliente:

Servidor: ***iperf –s –u –l 1472***
Cliente: ***iperf –c 10.80.47.251 -u –b 3m –l 1472 –t 15–i 3***

Los resultados de esta transmisión en el servidor son estos:

Intervalo(s)	Transferido (MBytes)	Velocidad(Mb/s)	Jitter (ms)	Datagramas perdidos
0.0-19.8	1,73	0,736	58,314	0 de 1234

Tabla 11. Resultado PANU-PANU con iperf con 1472 bytes de datos

Sin embargo, en el cliente aparece que el intervalo es de 0 a 15.4 segundos y que la velocidad es de 0,944 Mbits/s. Se produce una pérdida de velocidad debido al paso por el NAP y las dos conexiones L2CAP que existen.

Se observa que la velocidad es de 0,736 Mbits/s que es aproximadamente el 50% de la velocidad en el caso PANU-NAP que era 1,53 Mb/s. A su vez es algo inferior a la velocidad PANU-PANU pero con los paquetes ajustados a los paquetes DH5 que era de 0,863 Mbits/s.

6.3.3. Transferencias FTP

Para completar las medidas de rendimiento se añade una medida con un equipo de la Internet pública que usa el protocolo de transporte TCP. Para ello se va a medir el thoughput de la descarga de un fichero desde el servidor FTP anónimo "speedtest.tele2.net". Este servidor se utiliza precisamente para hacer este tipo de test. Para conectarse hay que ejecutar el comando ***ftp –p speedtest.tele2.net*** en el terminal de Linux.

La ***p*** indica que se usa el modo pasivo. En el modo activo se abre una conexión para datos desde el servidor hacia el cliente, es decir, una conexión desde fuera hacia dentro, lo que podría provocar que si el cliente estuviera protegido con un cortafuegos, éste filtrara o bloqueara la conexión entrante, al considerarlo un proceso desconocido. En el modo pasivo es el cliente quien inicia ambas conexiones, tanto control como datos, con lo que el cortafuegos ya no tiene ninguna conexión entrante que filtrar o bloquear.

Una vez conectado al servidor FTP, el acceso es mediante el usuario ***anonymous*** y no hace falta contraseña. Después se ejecuta el comando ***dir*** para ver los ficheros que se pueden descargar.

En la figura número 74 aparece el proceso de conexión al servidor ftp y se muestran los ficheros .zip con tamaños desde 1 KB hasta 100 GB.

```
local8-17@local8:~$ ftp -p speedtest.tele2.net
Connected to speedtest.tele2.net.
220 (vsFTPd 2.3.5)
Name (speedtest.tele2.net:local8-17): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
227 Entering Passive Mode (90,130,70,73,92,106).
150 Here comes the directory listing.
-rw-r--r--    1 0        0        1073741824000 Feb 19  2016 1000GB.zip
-rw-r--r--    1 0        0        107374182400 Feb 19  2016 100GB.zip
-rw-r--r--    1 0        0          102400 Feb 19  2016 100KB.zip
-rw-r--r--    1 0        0        104857600 Feb 19  2016 100MB.zip
-rw-r--r--    1 0        0        10737418240 Feb 19  2016 10GB.zip
-rw-r--r--    1 0        0        10485760 Feb 19  2016 10MB.zip
-rw-r--r--    1 0        0        1073741824 Feb 19  2016 1GB.zip
-rw-r--r--    1 0        0            1024 Feb 19  2016 1KB.zip
-rw-r--r--    1 0        0         1048576 Feb 19  2016 1MB.zip
-rw-r--r--    1 0        0        209715200 Feb 19  2016 200MB.zip
-rw-r--r--    1 0        0        20971520 Feb 19  2016 20MB.zip
-rw-r--r--    1 0        0         2097152 Feb 19  2016 2MB.zip
-rw-r--r--    1 0        0         3145728 Feb 19  2016 3MB.zip
-rw-r--r--    1 0        0        524288000 Feb 19  2016 500MB.zip
-rw-r--r--    1 0        0        52428800 Feb 19  2016 50MB.zip
-rw-r--r--    1 0        0          524288 Feb 19  2016 512KB.zip
-rw-r--r--    1 0        0         5242880 Feb 19  2016 5MB.zip
drwxr-xr-x    2 105      108          12288 May 31 09:24 upload
226 Directory send OK.
```

Figura 74. Conexión al servidor FTP

Para descargar el fichero se ejecuta el comando ***get*** con el nombre del archivo. Para descargar por ejemplo el archivo de 1MB se pone el comando ***get 1MB.zip***

En la siguiente figura se ve que el throughput aparece una vez descargado el fichero y se mide en KB/s. Se prueba descargando varios archivos de diferentes tamaños (1 MB, 3MB, 5MB, 10MB, 50MB).

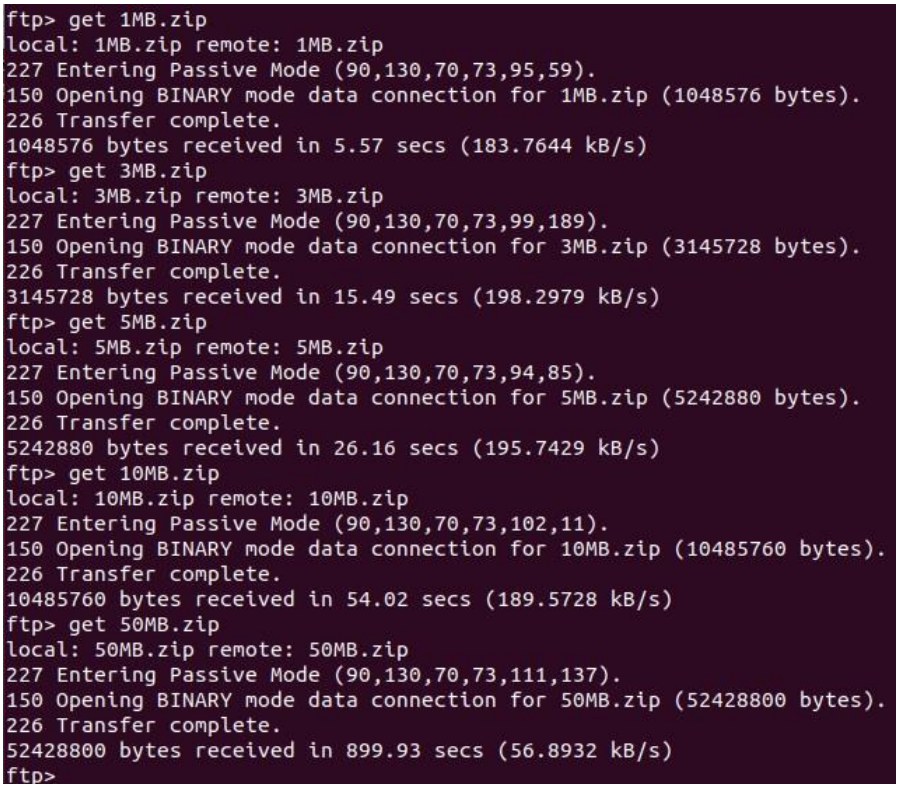

Figura 75. Throughput de las descargas FTP

Se realizan los cálculos para pasar los throughputs de descarga de KBytes/s a Mbits/s:

$$\textbf{Throughput\_1MB} = \frac{183{,}76\ \text{Kbytes x } 1024 \text{ x } 8}{1000000} = 1{,}505\ \text{Mbits/s}$$

$$\textbf{Throughput\_3MB} = \frac{198{,}29\ \text{Kbytes x } 1024 \text{ x } 8}{1000000} = 1{,}624\ \text{Mbits/s}$$

$$\textbf{Throughput\_5MB} = \frac{195{,}74\ \text{Kbytes x } 1024 \text{ x } 8}{1000000} = 1{,}603\ \text{Mbits/s}$$

$$\textbf{Throughput\_10MB} = \frac{189{,}57\ \text{Kbytes x } 1024 \text{ x } 8}{1000000} = 1{,}552\ \text{Mbits/s}$$

$$\textbf{Throughput\_50MB} = \frac{193{,}62\ \text{Kbytes x } 1024 \text{ x } 8}{1000000} = 1{,}586\ \text{Mbits/s}$$

Al tratarse de tráfico TCP, el tamaño máximo del segmento se ajusta a MSS que se utiliza en Ethernet (MTU = 1500 bytes) por tanto se producirá de nuevo segmentación en la capa banda base de Bluetooth.

Capítulo 7. Conclusiones y líneas futuras de trabajo

Se ha conseguido implementar el escenario basado en un punto de acceso a Internet del perfil PAN en el laboratorio docente de telemática en la ETSIIT. Una vez desplegado el escenario se ha hecho una campaña de medidas para la caracterización del rendimiento del protocolo IP y se ha comparado con los valores teóricos de acuerdo con la especificación Bluetooth.

Para que la configuración sea estable independientemente de las actualizaciones propias de un sistema operativo y de que haya que tener los permisos de administrador, se ha realizado en una máquina virtual Ubuntu de versión 17.10 (última versión disponible en el momento de la realización del trabajo). En la actualidad ya está disponible la versión 18.04 LTS que será mantenida hasta abril de 2023 por lo que una línea futura podría ser actualizar la máquina virtual a esta versión.

Otra línea de trabajo futura puede ser la de implementar el escenario GN ya que he leído en diversos foros que en las actuales versiones había algún problema a la hora de establecer los roles en dicha configuración y problemas con las versiones de BlueZ.

Por último, también se podría caracterizar el rendimiento del protocolo IP en una topología Scatternet, para ver los cambios que habría con respecto a este proyecto realizado en una piconet.

Bibliografía

[1] Sergio Alberto González Vergara, “Tecnología Bluetooth” Agosto de 2008.

http://tesis.ipn.mx/handle/123456789/6868

[2] R. Friedman, A. Kogan, Y. Krivolapov, "On Power and Throughput Tradeoffs of WiFi and Bluetooth in Smartphones," IEEE Transactions on Mobile Computing, vol. 12, no. 7, pp. 1363-1376, Julio 2013

[3] M.J. Morón Fernández, “Estudio del Rendimiento de Perfiles Bluetooth en Redes de Área Personal”, tesis doctoral, Departamento de Tecnología Electrónica E.T.S.I. de Telecomunicación, Universidad de Málaga, Abril 2008

[4] Janne M. Hagen and Vinh Pham, “An introduction to the Bluetooth technology and its applications” Norwegian Defence Research Establishment (FFI) 21 May 2015

[5] Core Specification Version 5.0 Active 06 Dec 2016

https://www.bluetooth.com/specifications/bluetooth-core-specification

[6] RFCOMM Version 1.2 Active 06 November 2012

https://www.bluetooth.com/specifications/protocol-specifications

[7] Serial Port Profile Version 1.2 Active 24 Jul 2012

https://www.bluetooth.com/specifications/profiles-overview

[8] Generic Object Exchange Profile Version 2.1.1 Active 15 Dec 2015

https://www.bluetooth.com/specifications/profiles-overview

[9] File Transfer Profile Version 1.3.1 Active 15 Dec 2015

https://www.bluetooth.com/specifications/profiles-overview

[10] Hands-Free Profile Version 1.7.1 Active 15 Dec 2015

https://www.bluetooth.com/specifications/profiles-overview

[11] Headset Profile Version 1.2Active 18 Dec 2008

https://www.bluetooth.com/specifications/profiles-overview

[12] Advanced Audio Distribution Profile Version 1.3.1 Active 14 Jul 2015

https://www.bluetooth.com/specifications/profiles-overview

[13] Video Distribution Profile Version 1.1 Active 24 Jul 2012

https://www.bluetooth.com/specifications/profiles-overview

[14] Bluetooth Network Encapsulation Protocol Version 1.0 Active 20 February 2003

https://www.bluetooth.com/specifications/protocol-specifications

[15] Personal Area Networking Profile Version 1.0 Active 20 Feb 2003

https://www.bluetooth.com/specifications/profiles-overview

[16] Última version de Ubuntu disponible

https://www.ubuntu.com/download/desktop

[17] J.M. Ruiz Ruiz de Villa, “Despliegue de una red IP multisalto basada en Bluetooth”, proyecto fin de carrera, ETSII y Telecomunicación, Universidad de Cantabria, 2004

[18] Solución para añadir las interfaces Wireshark

https://osqa-ask.wireshark.org/questions/1949/wireshark-says-there-are-no-interfaces-on-which-a-capture-can-be-done-how-do-i-fix-this

[19] Solución para arreglar el error sdptool browse local

https://bbs.archlinux.org/viewtopic.php?id=201672

Anexo

Se adjunta la hoja de características del dongle Bluetooth de Targus:

Targus

Targus Bluetooth 4.0 Dual-Mode Micro USB Adapter

ACB75AU

Overview

The Targus Bluetooth 4.0 Dual-Mode Micro USB Adapter allows you to use Bluetooth devices with laptops or desktops that are not Bluetooth enabled. Simply plug the adapter into your USB port and easily transfer files, use a Bluetooth mouse, printer, keyboard, keypad and more. The Bluetooth 4.0 Dual-Mode Micro USB Adapter provides wireless connectivity from up to 33 feet away freeing users from messy cables for a clutter-free workstation. With its ultra-mini, sleek design there is no need to remove the adaper in between uses, making it ideal for users to connect and forget. Equipped with Enhanced Data Rate, this Bluetooth 4.0 adapter allows wireless communication up to 3 times faster than Buetooth 1.X adapters.

Features

- Supports up to 10 metre range
- Transfer rate is 3x faster than Bluetooth 1.X adapters
- Connect up to 7 Bluetooth enabled devices such as smartphones, keyboards, mice & printers without the hassle of cables

Specifications

Warranty	Limited 1-Year Warranty
Other	USB Port
Other	Profiles supported: A2DP, AVCTP, AVDTP, AVRCP, BIP, BPP, DUN-DT, DUN-GW, FAX, FT, GAP, GAVDP, HCRP, HSP, HFP-AG, HID, OPP, PAN, SDP (Win7, Vista only: generic Attribute, Find Me Profile, Proximity Profile)
System Requirements	Windows 8, 7, XP, Vista
Weight	0.63kg

www.targus.com/au