ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

IMPLEMENTACIÓN DE *HONEYPOT* EN MÓDULO VIRTUAL PARA SEGURIDAD EN REDES EPN-ESFOT

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN ELECTRÓNICA Y TELECOMUNICACIONES

José Adrián Jibaja Robles
jose.jibaja01@epn.edu.ec

DIRECTOR: ING. FERNANDO VINICIO BECERRA CAMACHO, MSc.
fernando.becerrac@epn.edu.ec

CODIRECTOR: ING. FABIO MATÍAS GONZÁLEZ GONZÁLEZ, MSc.
fabio.gonzalez@epn.edu.ec

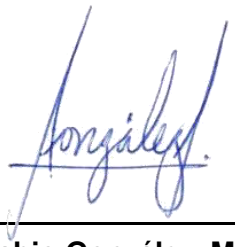
Quito, diciembre 2021

CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por el Sr. Jibaja Robles José Adrián como requerimiento parcial a la obtención del título de TECNÓLOGO EN ELECTRÓNICA Y TELECOMUNICACIONES, bajo nuestra supervisión:

Ing. Fernando Becerra, MSc.

DIRECTOR DEL PROYECTO



Ing. Fabio González, MSc.

CODIRECTOR DEL PROYECTO

DECLARACIÓN

Yo, Jibaja Robles José Adrián con CI: 1726499658 declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y que he consultado las referencias bibliográficas que se incluyen en este documento.

Sin prejuicio de los derechos reconocidos en el primer párrafo del artículo 144 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación – COESC-, soy titular de la obra en mención y otorgo una licencia gratuita, intransferible y no exclusiva de uso con fines académicos a la Escuela Politécnica Nacional.

Entrego toda la información técnica pertinente, en caso de que hubiese una explotación comercial de la obra por parte de la EPN, se negociará los porcentajes de los beneficios conforme lo establece la normativa nacional vigente.

José Adrián Jibaja Robles

DEDICATORIA

Dedico el presente trabajo a mi familia, quienes han sido el motor fundamental a lo largo de toda mi vida. En especial a mi mamá Graciela Elizabeth Robles Vélez y a mi hermano Javier Adolfo Jibaja Robles quienes con su guía, enseñanza y esfuerzo marcaron los lineamientos y valores en la búsqueda de las metas previstas.

José

AGRADECIMIENTO

Agradezco a mis padres, por brindarme la vida y ser los mentores más importantes de mi vida, permaneciendo siempre cerca de mí.

Doy gracias a mi madre, por ser el motor fundamental en el desarrollo de mi formación académica y apoyo incondicional de mi vida,

A mis hermanos, quienes me han acompañado y me han brindado su apoyo y compresión en todo momento.

A mi hermano Javier, quien supo enseñarme el valor del esfuerzo y encaminarme en la vida, su memoria siempre se mantendrá en mi corazón.

A la Escuela Politécnica Nacional por permitirme realizar mi desarrollo y formación profesional dentro de sus instalaciones.

Por último, doy gracias a cada uno de los profesores y docentes que han marcado las etapas más importantes de mi vida, con sus experiencias y enseñanzas en mi formación. Del mismo modo, expreso un profundo agradecimiento al director y codirector del proyecto.

José

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

RESUMEN

El presente proyecto de titulación consiste en la implementación de un módulo virtual para el desarrollo práctico en la asignatura de Seguridad en Redes en la carrera de Tecnología Superior en Redes y Telecomunicaciones. El módulo fue desarrollado bajo el sistema de seguridad *Honeypot*, con el objetivo de realizar prácticas de laboratorio en temas relacionados al manejo de sistemas operativos virtuales, configuración de protocolos y simulación de servicios.

En la primera sección se muestra la introducción, referente al desarrollo y planteamiento de los objetivos del proyecto. Se encarga de abordar y explicar los mecanismos que conforman al módulo virtual en base a la sustentación académica. Además, se establece la delimitación del alcance del proyecto.

En la segunda sección se muestra la metodología, referente a la explicación resumida del procedimiento de los lineamientos de la construcción del módulo. Se encarga de explicar la estructura de las hojas guías de las prácticas de laboratorio correspondiente a los docentes y estudiantes.

En la tercera sección se presentan los resultados, cuyo desarrollo y análisis se fundamenta en el entorno computacional, manejo de virtualización e interacción de la herramienta de seguridad *Honeyd*. De igual forma, se detalla la validación de las hojas guías de prácticas de laboratorio.

En la cuarta sección se encuentran las conclusiones y recomendaciones, referente a los resultados obtenidos en el desarrollo del proyecto.

En conclusión, el apartado final hace alusión a las referencias bibliográficas implementadas en el trabajo de titulación.

PALABRAS CLAVE: virtualización, *Honeypots*, *Honeyd.*

ABSTRACT

The present degree project consists of the implementation of a virtual module for practical development in the subject of Network Security in the Superior Technology in Networks and Telecommunications career. The module was developed under the Honeypot security system, with the objective of carrying out laboratory practices on topics related to the management of virtual operating systems, protocol configuration and service simulation.

The first section shows the introduction, referring to the development and approach of the project objectives. It is responsible for addressing and explaining the mechanisms that make up the virtual module based on academic support. In addition, the delimitation of the scope of the project is established.

The second section shows the methodology, referring to the summarized explanation of the procedure of the guidelines for the construction of the module. It is responsible for explaining the structure of the guide sheets of the corresponding laboratory practices to teachers and students.

The third section presents the results, which development and analysis is based on the computational environment, virtualization management and interaction of the Honeyd security tool. Similarly, the validation of the laboratory practice guide sheets is detailed.

The fourth section contains the conclusions and recommendations, referring to the results obtained in the development of the project.

In conclusion, the final section refers to the bibliographic references implemented in the degree work.

KEYWORDS: *virtualization, Honeypots, Honeyd.*

1 INTRODUCCIÓN

El avance de la tecnología en el mundo de las telecomunicaciones se ha ido incrementando de manera acelerada, esto debido al crecimiento de las infraestructuras de la red de datos, manejo de equipos, sistemas operativos y servicios sobre redes. Sin embargo, toda red de comunicación esta propensa a ser vulnerada debido a agentes externos o por elementos propios de la red, lo cual genera inconvenientes en la seguridad.

El limitado conocimiento sobre el tema y los sistemas de seguridad inexistentes son factores que contribuyen al establecimiento de un sistema vulnerable. Además, la aparición de “crackers” o “atacantes” dentro de una red, ocasionan diversos conflictos como errores del sistema, robos de información y daños sobre los equipos de cómputo. Según datos de un sitio web especializado en la información de la seguridad, cerca del 90% de los crackers cubren toda su actividad mediante la encriptación en elementos de red, razón por la cual no son percibidos a simple vista [1] [2]. Desde otro punto de vista, se puede encontrar diversos tipos de amenazas que intentan modificar el curso de la distribución de la red, mediante el mapeo de las direcciones IP, escaneo de puertos y manipulación de servicios [3].

Estos escenarios pueden ser reducidos o eliminados mediante el empleo de sistemas de seguridad “*Honeypots*”. Los *Honeypots* son sistemas que emplean la creación de *hosts* virtuales para la defensa de los sistemas principales mediante la simulación de estructuras de red [4].

El presente trabajo de titulación tiene el objetivo de proporcionar a la asignatura de Seguridad en Redes de la carrera de Tecnología Superior en Redes y Telecomunicaciones de la Escuela de Formación de Tecnólogos (ESFOT), un módulo virtual de un sistema de seguridad “*Honeypots*”, para el desarrollo de prácticas de seguridad en el manejo de estructuras de red, simulación de sistemas y manejo de servicios virtuales.

1.1 Objetivo general

Implementar un módulo virtual bajo la herramienta *Honeypot* para la simulación del desarrollo práctico en el tema de seguridad en la asignatura de Seguridad en Redes de la ESFOT.

1.2 Objetivos específicos

- Analizar los requerimientos funcionales del entorno de *Linux* en los laboratorios computacionales de la ESFOT.
- Interactuar en el entorno de *Linux* y su relación con los diferentes sistemas operativos en la máquina virtual mediante la herramienta *Honeyd*.
- Implementar un algoritmo de simulación de casos de protocolos, direcciones IP y *Host* mediante la herramienta *Honeyd*.
- Elaborar un mínimo de 4 guías prácticas para el desarrollo de actividades de laboratorio.
- Verificar el funcionamiento del módulo cuidando su calidad.

1.3 Fundamentos

Virtualización

Es un proceso de creación de un sistema operativo mediante el uso de software de una representación virtual de un recurso informático. Se realiza en base a una capa de abstracción del entorno físico, lo cual permite la división de elementos del sistema (memoria, almacenamiento, procesadores, etc.) con el objetivo de proporcionar sistemas virtuales, generalmente llamados máquinas virtuales (VM), cada uno es un computador independiente y se ejecuta en un espacio subyacente al sistema real [5] [6].

La estructura gráfica del sistema de virtualización se visualiza en la Figura 1.1.

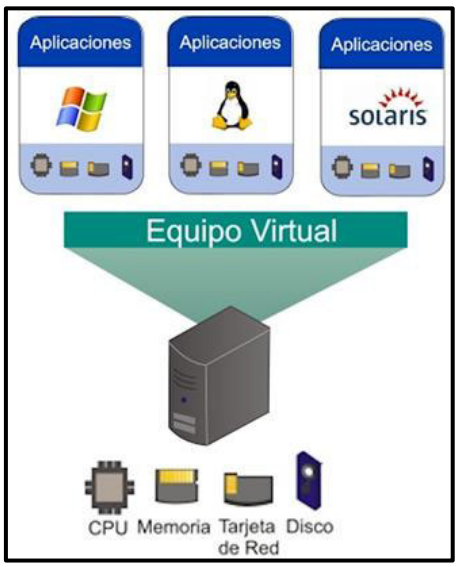


Figura 1.1 Sistema de virtualización [7]

Máquina Virtual

Es un mecanismo de simulación de un sistema físico o computador, generalmente conocido como “invitado” en el equipo físico “anfitrión”. Se caracteriza por manejar su propio sistema operativo (OS) y aplicaciones. Funciona en base a la aplicación de un software de virtualización o sistema de hipervisor [8] [9].

El diagrama de virtualización de máquinas se presenta en la Figura 1.2.

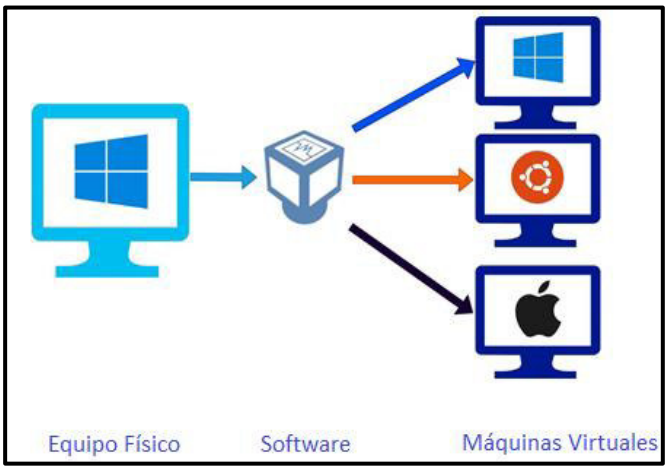


Figura 1.2 Virtualización de máquinas [10]

Hipervisor

Es un administrador de máquinas virtuales conocido como *Virtual Machine Manager* (VMM), encargado de relacionar al entorno virtual con el entorno físico. Es la capa de software focalizada en la coordinación y el uso compartido de recursos, memoria y procesamiento. Se encarga de mantener separada la existencia entre máquinas para evitar interferencia [9] [10] [11].

Tipos de hipervisores

Hipervisores nivel 1

Se denominan con el nombre de “*bare-metal*”, son aquellos que interactúan con los recursos físicos subyacentes al hardware. Se encargan de sustituir al sistema operativo por completo, además que su uso general se adjudica a los servidores virtuales de producción [11].

El diagrama de bloques del Hipervisor de nivel 1 se visualiza en la Figura 1.3.

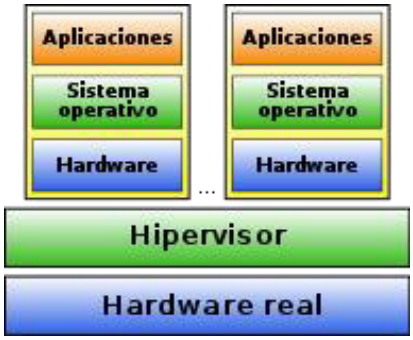


Figura 1.3 Hipervisor nivel 1 [12]

Hipervisores nivel 2

Se ejecutan como aplicaciones sobre un sistema operativo existente, enfocados en el manejo de sistemas operativos alternativos. Se caracterizan por el acceso en el sistema operativo principal y cuenta con una sobrecarga de rendimiento por los recursos físicos que son limitados y compartido con el sistema principal [11].

El diagrama de bloques del Hipervisor de nivel 2 se visualiza en la Figura 1.4.

Figura 1.4 Hipervisor nivel 2 [12]

Oracle VM VirtualBox

Es un software de código abierto desarrollado por *Oracle Corporation*, lanzado en el año 2007. Se caracteriza por el manejo de procesos de virtualización en arquitecturas *X86/amd64*. Se denomina como un hipervisor de tipo 2, encargado de los recursos virtuales y administración de máquinas virtuales en los equipos físicos [13].

El programa se encuentra disponible desde el sitio oficial: https://www.virtualbox.org/, ver en la Figura 1.5.

Figura 1.5 Sitio oficial de *Oracle VirtualBox* [13]

Unix

Es un sistema operativo que nació en los años 70, creado y desarrollado principalmente por Dennis Ritchie y Ken Thompson en los laboratorios *Bell* de *AT&T*. Se caracteriza por ser de código abierto (*Open Source*) y escrito en el lenguaje de programación C. Se encarga del desarrollo de un sistema operativo de múltiples tareas, programas y usuarios. Se basa en un sistema de archivos de orden jerárquico, en el uso de ficheros de acceso a carpetas y directorios [14] [15].

El sistema realiza la simulación del multiprocesamiento, la interconexión de procesos de comunicación de usuarios y conexión de periféricos. Se focaliza en la administración central de múltiples usuarios, redirección de entradas y salidas en la creación de software [15] [16].

GNU/Linux

Es un sistema operativo de código abierto tipo *Unix*, basado en herramientas básicas del proyecto GNU (*GNU's Not Unix*) anunciado por Richard Stallman en 1983 y sustentado por la FSF (*Free Software Foundation)*. La denominación del nombre de *Linux* se basa en el manejo del núcleo (*Kernel*) desarrollado por Linus Torvalds en 1991. Se caracteriza por el uso del software libre; el código fuente cuenta con libertades de uso, modificación y redistribución libre bajo los términos de la Licencia Pública General GLP [17].

Características

- **Estabilidad.-** Es un sistema estable y robusto debido a las configuraciones del nivel administrativo. Se encarga del manejo de las actualizaciones, descargas y reforzamiento del sistema. [18].
- **Multitarea y multiusuario.-** Se encarga de la ejecución segura de múltiples programas y el acceso a múltiples usuarios en simultáneo [18].

- **Seguridad.-** Es un sistema de gestión propia a nivel de usuario, consta de herramientas y actualizaciones de paquetes. Se encarga de la gestión de contraseñas, niveles de usuarios y permisos [18].
- **Distribuciones.-** Son las variantes del sistema operativo encargadas de satisfacer las necesidades específicas [18].

Ubuntu

Es una distribución del sistema operativo *GNU/Linux* de código abierto lanzado en el año 2004, basado en el sistema operativo *Debian.* Es desarrollado bajo el esquema del empresario Mark Shuttleworth, cuyo objetivo es promover y promocionar de manera gratuita el uso del software libre a usuarios promedios. Es usado en equipos de escritorio y servidores [19] [20].

Características

- **Terminal e interfaz.-** Se maneja en base a una terminal por medio de la línea de comandos similar a los sistemas *Unix*; creación, desarrollo y gestión de archivos. Además, cuenta con una interfaz gráfica intuitiva y personalizable [20].
- **Orientación de red.-** Se compone del uso de tarjetas de red, con el objetivo de realizar actividades relacionadas a la intercomunicación, manejo de red y validación de datos [20].
- **Compatibilidad.-** Su compatibilidad se relaciona en gran medida con el hardware existente, debido a los requisitos mínimos de instalación, tales como; procesador 2 (GHz), memoria 2 (GB) de RAM, disco duro 25 (GB) y conexión a Internet [20].
- **Sistema seguro.-** Es un sistema de naturaleza robusta, por lo que cuenta con actualizaciones periódicas para el control de dispositivos. Se caracteriza por proporciona un nivel alto de seguridad para empresas y organizaciones [20].

Kali Linux

Es una distribución del sistema operativo de *GNU/Linux* de código abierto lanzado en el año 2013, conocido antiguamente como *BackTrack Linux*. Se basa en los sistemas *Debian*, es fundada y mantenida por Mati Aharoni y Devon Kearns de la *Offensive Security*. Contiene una gran cantidad de herramientas focalizadas en el uso de seguridad informática y auditoría. Se caracteriza por la realización de pruebas de penetración, actualizaciones, servicios de red y análisis de vulnerabilidades. Es una

solución multiplataforma, accesible y disponible gratuitamente para profesionales de la seguridad de la información [21] [22].

Características

- **Herramientas de Pruebas de Penetración.-** Se dispone de una gran cantidad de herramientas (600), entre ellas se encuentra; *Nmap* (escáner de puertos), *Wireshark* (*sniffer* de paquetes de red), *John the Ripper* (*crackeador* de contraseñas), y *Aircrack-ng* (software de pruebas de seguridad de redes inalámbricas) [22].
- **Compatibilidad con FHS.-** Cuenta con el manejo del Estándar de Jerarquía del Sistema de Archivos FHS (*Filesystem Hierarchy Standard*). Normativo encargado de los directorios y archivos en los sistemas *GNU/Linux* [22].
- **Instalación.-** Se compone de requisitos mínimos de instalación; procesador 2 (GHz), memoria 4 (GB) de RAM, disco duro 8 (GB) y conexión a Internet [22].
- **Entorno seguro.-** Se dispone de un equipo de trabajo encargado de la actualización y manejo de paquetes en repositorios seguros por medio de múltiples protocolos de seguridad [22].
- **Paquetes y repositorios firmados por GPG.-** Se dispone de paquetes con cifrado y firmas digitales por medio del programa GPG (*GNU Privacy Guard*). Se enfoca en la autoría de los desarrolladores de los repositorios y paquetes verificados en los sistemas *GNU/Linux* [22].

Honeypot

Es una herramienta o mecanismo de seguridad informática de código abierto encargada de proporcionar un sistema trampa o señuelo sobre la red con el objetivo de evitar un posible ataque informático. Se caracteriza principalmente por detectar y obtener información del atacante. Además, se encarga de localizar el origen y posteriormente implementar acciones de seguridad [23].

Funciones principales

- Desviar las interacciones del atacante sobre la red y proteger los recursos principales del sistema activo [23].
- Formar base de datos de perfiles de los atacantes y construir medidas de prevención [23].
- Descubrir posibles vulnerabilidades de los sistemas operativos, programas y entornos que no se encuentren debidamente documentados [23].

Clasificación de *Honeypots*

Se clasifican en función del ambiente de operación**:**

- **Producción.-** Son los encargados de proteger los ambientes reales con servicios vulnerables o puertos abiertos en equipos de uso diario [24].
- **Investigación.-** Son los encargados de proporcionar el material educativo o investigativo, con el objetivo principal de estudiar los patrones de ataque y prevenir futuras amenazas [24].
- **Interacción.**- Son aquellos que se disponen de acuerdo a un nivel de interacción y rango de protección [24].

Tipos de *Honeypots* por el nivel de interacción

Honeypots de baja interacción

Son sistemas de interacción en un medio virtual que se encargan de simular sistemas operativos, aplicaciones y servicios abiertos. Se caracterizan por limitar la actividad del atacante al entorno de emulación. Esto genera un menor riesgo y mayor simplicidad de implementación, debido a que los servicios emulados limitan el riesgo de penetración al restringir el acceso al sistema operativo real. Además, son sistemas que cuentan con facilidad de uso y mantenimiento mínimo [24] [25].

La captura de datos de los ataques es controlada, lo que implica una capacidad de registro limitada. La simulación de servicios cuenta con un límite de operatividad. Sin embargo, el atacante visualiza un sistema real [25] [26].

La representación gráfica de un *Honeypot* de baja interacción se visualiza en la Figura 1.6.

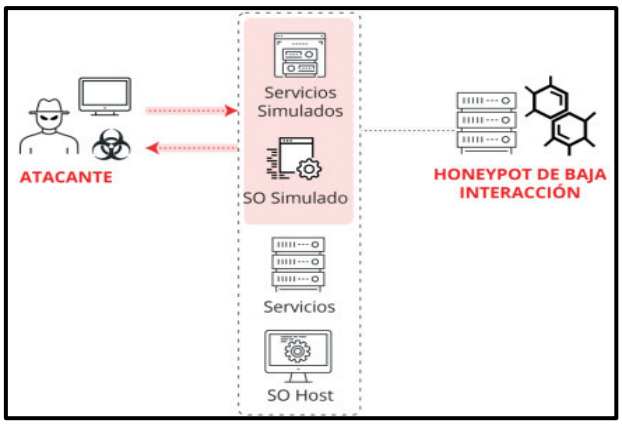


Figura 1.6 *Honeypot* de baja interacción [26]

Honeypots de alta interacción

Son sistemas de interacción en un medio real que se encargan de levantar sistemas operativos, aplicaciones y servicios en un ordenador o una red troncal. Se caracterizan por un entorno de funcionamiento exterior o real, lo que conlleva a establecer un sistema perfectamente protegido. Se encargan de establecer un sistema operativo realista sobre los atacantes, permitiendo recabar información para el manejo del análisis de operación y estudiar nuevas formas de comportamiento [24] [25] [26].

Se componen de una mayor capacidad de almacenamiento de información, permiten una interacción completa del atacante sobre los equipos señuelo. El sistema de manejo de *Honeypots* de alta interacción sobre redes se conoce bajo la denominación del proyecto *Honeynet* [24] [25] [26].

La representación gráfica de un *Honeypot* de alta interacción se visualiza en la Figura 1.7.

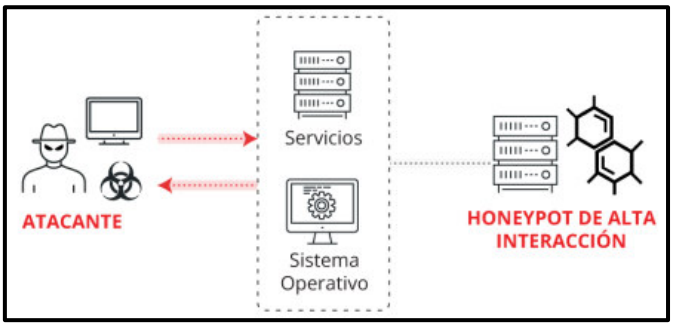


Figura 1.7 *Honeypot* de alta interacción [26]

Honeyd

Es un *Honeypot* de baja interacción desarrollado por Niels Provos encargado de crear y ejecutar múltiples *Honeypots* virtuales sobre una red informática. Se caracteriza por la configuración de personalidades en base a la imitación de sistemas operativos y emulación de servicios. Se encuentra disponible como un software de código abierto lanzado bajo la licencia GPL y compatible con los sistemas *Unix, GNU/Linux y Windows*. [27] [28].

El sistema de *Honeyd* se encarga de reenviar el tráfico hacia *Honeypots* a través de un enrutador o mediante el Protocolo de Resolución de Direcciones (*Address Resolution Protocol)* (ARP). Los *Honeypots* se encargan de simular un sistema operativo diferente sobre la red [28].

La topología de funcionamiento de un sistema *Honeyd* se visualiza en la Figura 1.8.

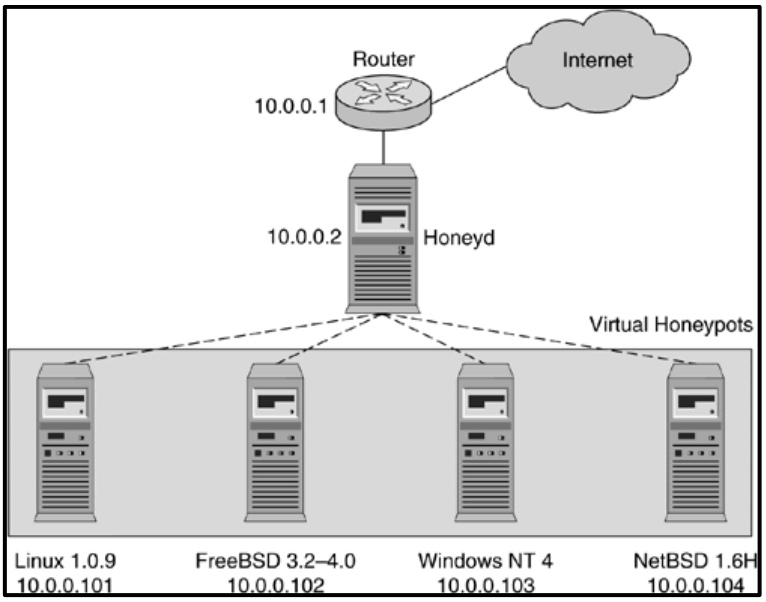


Figura 1.8 Topología de un sistema *Honeyd* [28]

Características

- **Simulación de *Honeypots* virtuales.-** Se dispone de una amplia capacidad de simulación, de hasta 65536 equipos virtuales en simultáneo. Se enfoca en la interacción individual o colectiva de cada *host*, permitiendo comportamientos específicos de *host* [28].
- **Emulación de servicios.-** Cuenta con un proceso de emulación de servicios en relación a un archivo de configuración. Además, se encarga de notificar los eventos de conexión, entre el origen y el destino, estableciendo el señuelo con el atacante [28].
- **Simulación de sistemas operativos.-** Maneja el desarrollo de la simulación de los sistemas operativos mediante el modelo TCP/IP, por medio de la configuración de personalidades mediante la asignación de direcciones del Protocolo de Internet (*Protocol Internet*) IP, puertos de capa de transporte del Protocolo de Control de Transmisión (*Transmission Control Protocol*) TCP y el Protocolo de Datagramas de Usuario (*User Datagram Protocol*) UDP, control de capa de red por medio del Protocolo de Mensajes de Control de Internet (*Internet Control Message Protocol*) ICMP y servicios en cada equipo virtual [28].
- **Simulación de topología de enrutamiento.-** Se dispone en simulación de topologías de enrutamiento, permitiendo configurar parámetros de red, tales como la latencia, pérdida de paquetes y ancho de banda. Además, permite la integración de máquinas físicas con topologías virtuales [28].

- **Subsistema de virtualización.-** Dispone de la ejecución de aplicaciones reales del tipo *Unix* con nombres virtuales; servidores web, FTP, entre otros. Además, permite la unión de puertos dinámicos en direcciones virtuales permitiendo la iniciación de conexiones de red [28].

Nmap

Es un software de código abierto multiplataforma desarrollado por Gordon Lyon, lanzado en el año 1997. Se caracteriza por el proceso de evaluación del uso de la seguridad en sistemas informáticos. Se encarga del envío y recepción de paquetes definidos para el análisis de resultados de la información relacionada a equipos activos y servicios sobre la red [29].

Se enfoca en el desarrollo de multifunciones orientadas al sondeo de redes de computadoras, detección de equipos, análisis de servicios y manejo de sistemas operativos. Las funciones se extienden en base al uso de *scripts* o programas de establecimiento de servicios enfocados a la configuración avanzada de las vulnerabilidades y aplicaciones. Además, el proceso de escaneo se adapta a las configuraciones de la red, de acuerdo con la congestión y latencia producida [29].

Funciones Principales

- Distribuido en base al desarrollo de software libre bajo la Licencia Pública General (GLP) [29].
- Descubrir ordenadores, servidores y equipos de red en base a la implementación de herramientas de localización (*ping*) [29].
- Identificar puertos (TCP/UDP) abiertos dentro de un equipo u ordenador objetivo [29].
- Determinar servicios (puertos) activos en equipo u ordenador objetivo [29].
- Determinar el sistema operativo (OS) y versión del equipo siendo la Huella Dactilar (*Fingerprinting*) [29].
- Recolectar características de hardware de equipos u ordenador objetivo [29].

2 METODOLOGÍA

2.1 Descripción de la metodología usada

Para el desarrollo del módulo virtual fue necesario determinar las condiciones físicas y características de funcionamiento para el empleo de los equipos computacionales. Por

lo que se realizó el proceso de investigación con el fin de localizar y determinar los elementos disponibles con los requerimientos mínimos para el desarrollo y la implementación.

En la primera fase se realizó el análisis de los requerimientos sobre la disposición del sistema operativo *Linux* en los recursos tecnológicos de la ESFOT. La asignación del sistema operativo de *GNU/Linux* se consideró por las características del sistema, siendo una plataforma de código abierto encargada del manejo, clasificación, organización y ejecución de programas en la línea de comando para la resolución de los sistemas operativos en entornos virtuales.

Por lo cual, se procedió a realizar la localización del sistema mediante la coordinación con la dirección de la ESFOT, a través del jefe de laboratorios de TIC para determinar la disposición del sistema. El planteamiento del módulo fue evaluado en base al funcionamiento de los sistemas de virtualización, con el objetivo de manejar varios sistemas operativos al mismo tiempo.

La plataforma seleccionada y disponible en los recursos tecnológicos fue el software de virtualización de *Oracle VirtualBox* en su versión 6.0, la misma que se puso a disposición para la descarga, instalación y manejo de las distribuciones de *GNU/Linux* de *Ubuntu* y *Kali Linux*, las cuales se relacionan a la estructura del módulo virtual.

La disposición de los equipos y el entorno de trabajo fue asignada en base a la cantidad de estudiantes inscritos en la asignatura de Seguridad en Redes. En el análisis se detalló la cantidad de estudiantes por periodo, por lo que la máxima autoridad de los laboratorios designó el uso de 37 equipos de cómputo en el Laboratorio Marcelo Dávila para el desarrollo de las actividades.

En la segunda fase se realizó el proceso de interacción con *Linux* para el manejo del software de *Honeyd* en la descripción de los sistemas operativos. El manejo de los sistemas operativos sobre el uso del software libre se consideró debido a la capacidad de manejar programas para el desarrollo virtual, por tal motivo en el módulo se buscó el uso de sistemas trampa enfocados en la seguridad informática.

Debido a esto, se eligió a los sistemas de seguridad de *Honeypots*, los cuales se caracterizan por establecer sistemas reales o señuelos.

Por tal motivo, se designó al *Honeypot* de *Honeyd* para el desarrollo de la simulación de sistema a través de la detección, protección y registro de información del atacante. En

el módulo se realizó una interacción con *Honeyd* con la finalidad de conocer los sistemas operativos y funciones principales, para llevar a cabo el proceso se requirió realizar la fase de instalación, análisis de componentes y comprobación de resultados mediante pruebas de funcionamiento.

La fase de instalación se realizó en el sistema operativo de *Ubuntu* mediante la creación del paquete de *Honeyd*. La creación se desarrolló por medio de la descarga de librerías, descompresión de archivos y la ejecución de archivos de las librerías de soporte de *Honeyd*.

La fase de análisis se designó en el directorio principal de *Honeyd* con la descripción del archivo de configuración "*honeyd.conf*" y el archivo de información "*README*". La fase de pruebas de funcionamiento se realizó con inicialización de los archivos "*scripts*", relacionados al servicio de ARP en el manejo de tráfico de red y al servicio *Honeyd* relacionado al otorgamiento de los comandos iniciales.

De igual manera, se procedió a describir los archivos de la base de datos de nombres de personalidades y sistemas operativos, siendo "*Nmap.prints*" y "*Nmap.assoc*", referentes a las identidades de los *Honeypots*, describiendo los sistemas operativos más relevantes: *Windows, MAC OS y Linux*.

En la tercera fase se realizó el manejo de la herramienta de *Honeyd* a través del archivo de configuración "*honeyd.conf*". Se procedió a analizar el software de *Honeyd* en base al diagrama de bloques, el cual describe en forma de procesos el manejo del archivo de configuración, explicando el funcionamiento de cada paquete simulado a través de las líneas de comando. Las cuales, se encargan de especificar la función individual de cada parámetro.

La fase de análisis se designó a la interpretación del algoritmo, debido a que el archivo de configuración y las líneas de comandos son el documento manipulable en el cual se desarrollan las configuraciones de los casos, como son la personalidad, los protocolos (TCP, UDP e ICMP), los puertos y las direcciones IP.

La fase de diseño se designó a la capacidad de cambio de nombre del archivo de configuración y la capacidad de agregar comentarios en forma de etiqueta. Los elementos de simulación y prueba de funcionamiento se detallaron en la sección de hojas guías.

En la cuarta fase se elaboró 4 hojas guías para prácticas de laboratorio que comprenden el manejo, configuración y funcionamiento del módulo virtual para que sean ejecutadas en la asignatura de Seguridad en Redes. Dichas prácticas contienen instrucciones tanto para los docentes como para los estudiantes. Los temas a ser tratados fueron:

Introducción a los *Honeypots*.- El desarrollo de la práctica se enfoca al establecimiento de las generalidades del módulo virtual, la familiarización del entorno, manejo de archivos principales, archivo de configuración básica, asignación de dirección IP, comandos básicos de inicialización, ejecución de *Honeypot* simple, verificación de creación en máquina externa y análisis de resultados en el archivo de registro de información.

Validación de puertos TCP y UDP.- El desarrollo de la práctica se enfoca en la configuración de los puertos (TCP y UDP), ejecución de *Honeypot*, validación de conexión, escaneo de puertos (TCP y UDP) en máquina externa y análisis de resultados en el archivo de registro de información.

Acciones de protocolos TCP, UDP e ICMP.- El desarrollo de la práctica se enfoca en la configuración de acciones de los protocolos (TCP, UDP e ICMP), ejecución de conjunto de *Honeypots*, escaneo de puertos (TCP y UDP) en *Honeypots*, resultados de interacción en relación al acceso y bloqueo de puertos en *Honeyd*, visualizado en máquina externa y análisis de resultados en el archivo de registro de información.

Simulación de servicios TCP.- El desarrollo de la práctica se enfoca en la configuración de la simulación de servicios en los puertos TCP, configuración de archivo "*scripts*" de simulación, ejecución de conjunto de *Honeypots*, validación de conexión, escaneo de puertos TCP*,* comprobación de servicios señuelos y análisis de resultados en el archivo de registro de información.

En la última fase se realizó el proceso de verificación referente al funcionamiento del módulo virtual. Se realizó la importación de las máquinas virtuales sobre los equipos computacionales y se verificó el funcionamiento correcto de cada una de las prácticas. Además, se desarrollaron videos demostrativos de la funcionalidad de las prácticas realizadas y se expusieron los procedimientos adecuados para la obtención de los resultados esperados.

3 RESULTADOS Y DISCUSIÓN

La implementación del módulo virtual tiene el objetivo de dar a conocer los sistemas de seguridad de *Honeypots* mediante la herramienta *Honeyd*. El objetivo es evidenciar el manejo de los puertos (TCP y UDP), direcciones IP y servicios mediante el uso de hojas guías para la contribución del desarrollo académico de los estudiantes y docente encargado de la asignatura de Seguridad en Redes en la carrera de Tecnología Superior en Redes y Telecomunicaciones.

3.1 Análisis de requerimientos del laboratorio en equipos computacionales

El desarrollo del presente proyecto se enfoca en el uso del Laboratorio Marcelo Dávila ubicado en el Edificio No. 21 en la Dirección de la Escuela de Formación de Tecnólogos (ESFOT) de la Escuela Politécnica Nacional (EPN). La asignación del laboratorio se realizó en base a la coordinación y orientación por parte del Ing. William Nacimba (william.nacimbac@epn.edu.ec), jefe encargado de los Laboratorios de TIC de la ESFOT.

Laboratorio Marcelo Dávila

Es un espacio académico en el área de las Tecnologías de la Información y Comunicación (TIC), disponible para el desarrollo de actividades del personal docente y estudiantes de la Escuela Politécnica Nacional. Se encarga de brindar recursos informáticos a nivel de laboratorio en impartición de clases, implementación de módulos y manejo de software [30].

La disposición física del laboratorio se visualiza en la Figura 3.1.

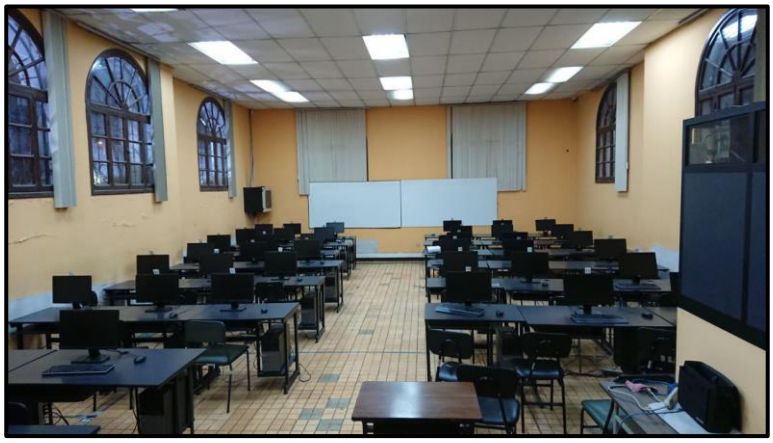


Figura 3.1 Laboratorio Marcelo Dávila [30]

Equipamiento

El entorno de trabajo se dispone del equipamiento de ordenadores de escritorio pertenecientes a la marca *DELL*, ver en la Figura 3.2.

Figura 3.2 Ordenador de escritorio marca *DELL* [30]

Las características y el número de ordenadores se encuentran descritos en la Tabla 3.1.

Tabla 3.1 Características del Laboratorio Sala Marcelo Dávila

Laboratorio Sala Marcelo Dávila	
Equipos	**Características**
37	**Sistema Operativo**: *Windows* 10 pro. **Procesador**: *Intel Core i7* de 4ta generación. **Memoria RAM:** 8 (GB) para 64 bits. **Espacio en Disco Duro**: 500 (GB) para 64 bits

Análisis de estudiantes en la asignatura de Seguridad en Redes

El proyecto se desarrolla en función a los estudiantes inscritos en la asignatura de Seguridad en Redes. En tal virtud, y en conjunto con la coordinación con el jefe del laboratorio se dispone a establecer el número de equipos de trabajo de acuerdo al margen de estudiantes inscritos en los periodos activos.

La información solicitada es proporcionada por la Subdirección de la ESFOT. Por tal motivo, el número de estudiantes inscritos en la asignatura se describe en la Tabla 3.2.

Tabla 3.2 Descripción de estudiantes en la asignatura de Seguridad en Redes

Código	Materia	Periodo	Estudiantes
TRTR512	Seguridad en Redes	2020-B	5
TRTR512	Seguridad en Redes	2021-A	24
TRTD542	Seguridad en Redes	2021-B	15

Designación de equipos

El análisis de los estudiantes presenta la cantidad y la variación de personas inscritas dentro de la asignatura, presentando al mayor número en el periodo 2021-A con 24 estudiantes.

Sin embargo, debido a las proyecciones o posibles eventualidades en el índice de aprobación, se acordó con el jefe del laboratorio disponer de la capacidad máxima de equipos disponibles en las instalaciones, con una cantidad de 37 equipos en total.

Las características de los equipos se describen en la Tabla 3.3.

Tabla 3.3 Equipos del entorno de trabajo

No.	Número de Ordenador	Marca/Modelo	Código
1	No.1	*DELL*/9020	17799990000-9014347
2	No.2	*DELL*/9020	17799990000-9014348
3	No.3	*DELL*/9020	17799990000-9014349
4	No.4	*DELL*/9020	17799990000-9014351
5	No.5	*DELL*/9020	17799990000-9014350
6	No.6	*DELL*/9020	17799990000-9014352
7	No.7	*DELL*/9020	17799990000-9014353
8	No.8	*DELL*/9020	17799990000-9014354
9	No.9	*DELL*/9020	17799990000-9014355
10	No.10	*DELL*/9020	17799990000-9014356
11	No.11	*DELL*/9020	17799990000-9014357
12	No.12	*DELL*/9020	17799990000-9014358
13	No.13	*DELL*/9020	17799990000-9014359
14	No.14	*DELL*/9020	17799990000-9014360
15	No.15	*DELL*/9020	17799990000-9014361
16	No.16	*DELL*/9020	17799990000-9014362

No.	Número de Ordenador	Marca/Modelo	Código
17	No.17	*DELL*/9010	17799990000-7195094
18	No.18	*DELL*/9020	17799990000-9014364
19	No.19	*DELL*/9020	17799990000-9014365
20	No.20	*DELL*/9020	17799990000-9014366
21	No.21	*DELL*/9010	17799990000-7195126
22	No.22	*DELL*/9010	17799990000-7195135
23	No.23	*DELL*/9010	17799990000-7195121
24	No.24	*DELL*/9010	17799990000-7195132
25	No.25	*DELL*/9010	17799990000-7195115
26	No.26	*DELL*/9010	17799990000-7195140
27	No.27	*DELL*/9010	17799990000-7195116
28	No.28	*DELL*/9010	17799990000-7195163
29	No.29	*DELL*/9010	17799990000-7195146
30	No.30	*DELL*/9010	17799990000-7195149
31	No.31	*DELL*/9010	17799990000-7195164
32	No.32	*DELL*/9010	17799990000-7195165
33	No.33	*DELL*/9010	17799990000-7195128
34	No.34	*DELL*/9010	17799990000-7195099
35	No.35	*DELL*/9010	17799990000-7195100
36	No.37	*DELL*/9010	17799990000-7195176
37	No.38	*DELL*/9010	17799990000-7195177

Requerimientos del sistema de virtualización

Los requerimientos mínimos para establecer el manejo del sistema de virtualización *Oracle VirtualBox*, se describen en la Tabla 3.4.

Tabla 3.4 Requerimientos del sistema de virtualización [31]

Requerimientos del sistema de virtualización	
Sistema Operativo	**Características**
Windows (7,8,10)	**Procesador**: Compatible con *Intel* y *AMD*. **Memoria RAM**: 4 (GB) **Espacio en Disco Duro**: 32 (GB)

Disposición del sistema de virtualización

El sistema de virtualización de *Oracle VirtualBox* se encuentra disponible dentro del Laboratorio Marcelo Dávila. El software de código abierto está instalado en la versión 6.0. En tal virtud, se procede a realizar la instalación, configuración y establecimiento de las máquinas virtuales pertenecientes al proyecto.

La validación del sistema de virtualización en el laboratorio se visualiza en la Figura 3.3.

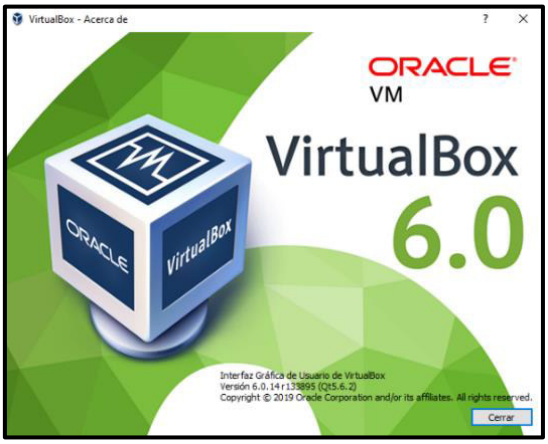


Figura 3.3 Sistema de virtualización en laboratorio

Clasificación de máquinas virtuales

Para el desarrollo del proyecto se procede con la ejecución de las dos máquinas virtuales enfocadas en el manejo del sistema operativo *GNU/Linux*. El establecimiento de las máquinas virtuales se realiza en base a la descripción de la Tabla 3.5.

Tabla 3.5 Desglose de máquinas del módulo virtual

No.	Sistema Operativo	Distribución
1	*GNU/Linux*	*Ubuntu 14.04*
2	*GNU/Linux*	*Kali Linux 2020.4*

Descarga del sistema operativo *Ubuntu 14.04.6*

Se realiza la descarga de la imagen ISO correspondiente al sistema operativo *Ubuntu 14.04.6*. La imagen ISO se encuentra disponible desde el repositorio *Mirror* EPN: https://mirror.epn.edu.ec/ubuntu-releases/14.04/. La descarga se realiza mediante la acción "*click*" sobre el nombre de la imagen ISO en la columna de "*Name"*, ver en la Figura 3.4.

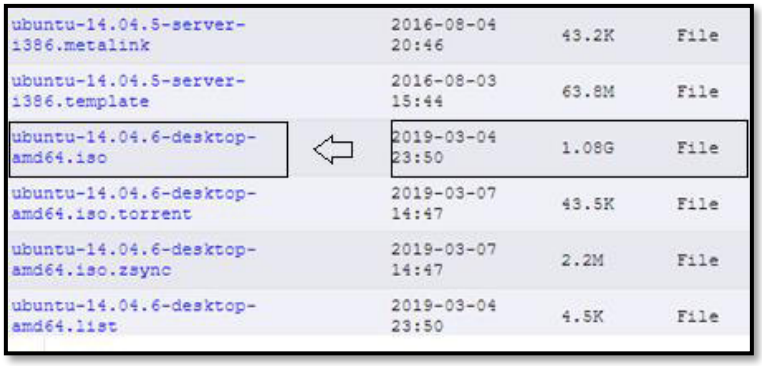


Figura 3.4 Descarga de imagen ISO de *Ubuntu* 14.04.6

Creación de *Ubuntu* en *Oracle VM VirtualBox*

Una vez descargada la imagen ISO de *Ubuntu 14.04.6* se procede a crear una máquina virtual para el sistema operativo.

Se debe ingresar a la herramienta *Oracle VM VirtualBox* y posteriormente seleccionar la opción "Nueva" para la creación de la máquina virtual, ver en la Figura 3.5.

Figura 3.5 Creación de máquina virtual *Ubuntu*

A continuación, se debe ingresar las características del equipo tales como el nombre, tipo de sistema y versión de trabajo. Además, se debe seleccionar la opción "*Next*" para continuar, ver en la Figura 3.6.

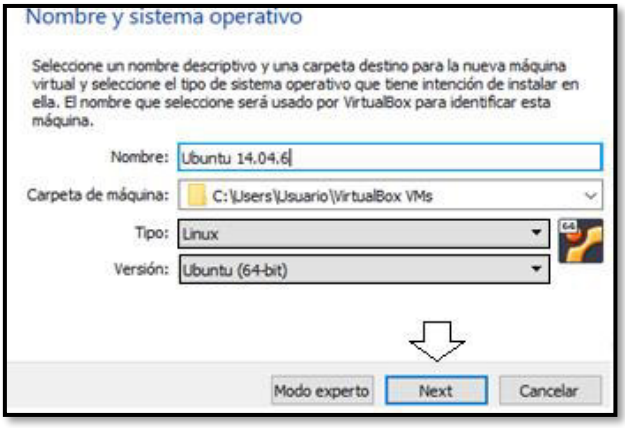


Figura 3.6 Características de máquina virtual *Ubuntu*

Se debe asignar el tamaño de la memoria RAM a un valor de un 1 (GB) como requerimiento básico del módulo y selecciona la opción "*Next*" para continuar, ver en la Figura 3.7.

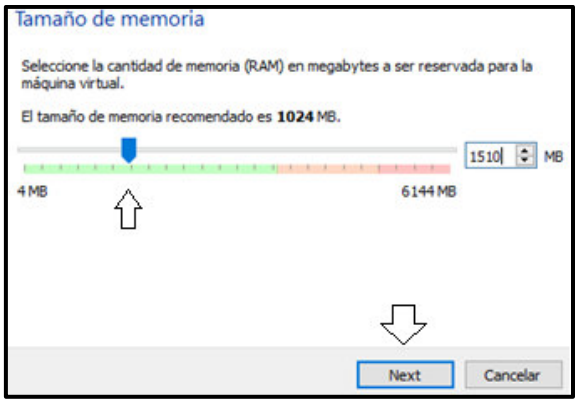


Figura 3.7 Asignación de memoria RAM en *Ubuntu*

Se debe seleccionar la opción "*Crear un disco duro virtual*" y la opción "Crear" para continuar, ver en la Figura 3.8.

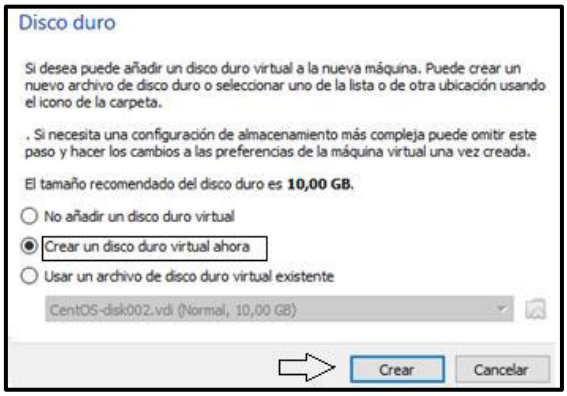


Figura 3.8 Configuración de disco duro virtual en *Ubuntu*

Se debe seleccionar la opción de "*VDI (VirtualBox Disk Image)* " y seleccionar "*Next*" para continuar, ver en la Figura 3.9.

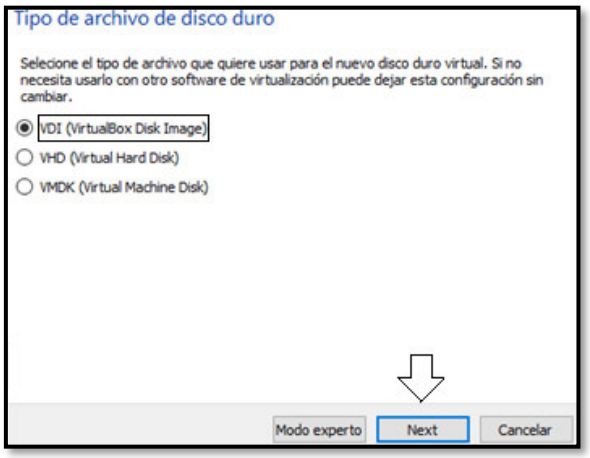


Figura 3.9 Selección de archivo de disco duro virtual en *Ubuntu*

Se debe seleccionar la opción de "*Reservado dinámicamente*" y seleccionar "*Next*" para continuar, ver en la Figura 3.10.

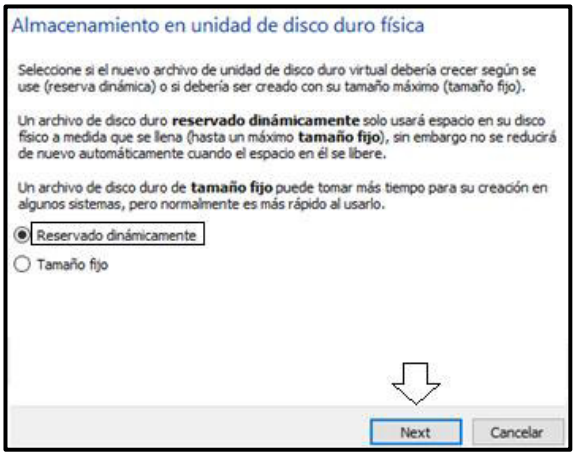


Figura 3.10 Adjudicación de almacenamiento dinámico en *Ubuntu*

Se muestra la ruta de ubicación de la máquina y el tamaño del disco duro. Se debe seleccionar la opción "Crear" para continuar, ver en la Figura 3.11.

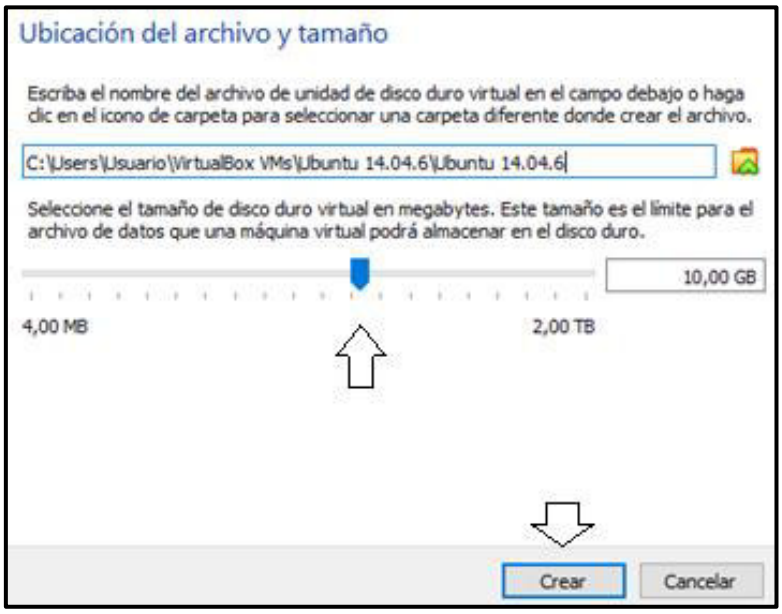


Figura 3.11 Ruta de ubicación de máquina virtual *Ubuntu*

Instalación de *Ubuntu 14.04.6*

Se debe ingresar a la opción de "*Configuración",* en el apartado de "*Sistema*" e ingresar a "*Procesador*", con el fin de aumentar el procesador de la máquina 2 *CPU*, ver en la Figura 3.12.

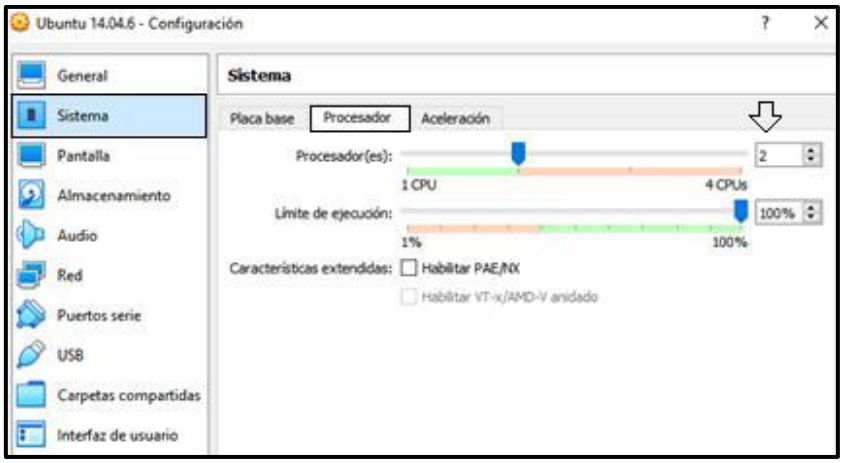


Figura 3.12 Configuración de procesador virtual en *Ubuntu*

Se debe dirigir a la opción de "*Pantalla*" en el apartado de "*Aceleración*" y habilitar la aceleración 3D, ver en la Figura 3.13.

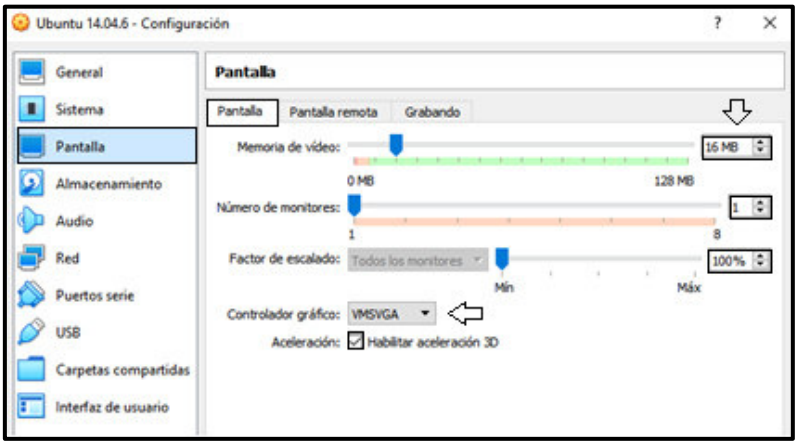


Figura 3.13 Configuración de pantalla virtual en *Ubuntu*

Se procede a ingresar a la carpeta de almacenamiento de imágenes ISO, ver en la Figura 3.14.

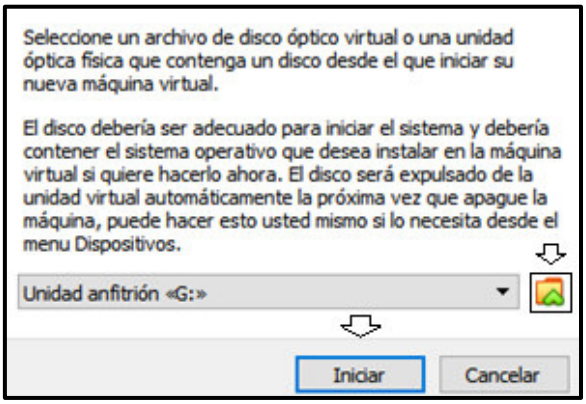


Figura 3.14 Carpeta de imágenes ISO

Se debe seleccionar la imagen ISO del sistema *Ubuntu*, ver en la Figura 3.15.

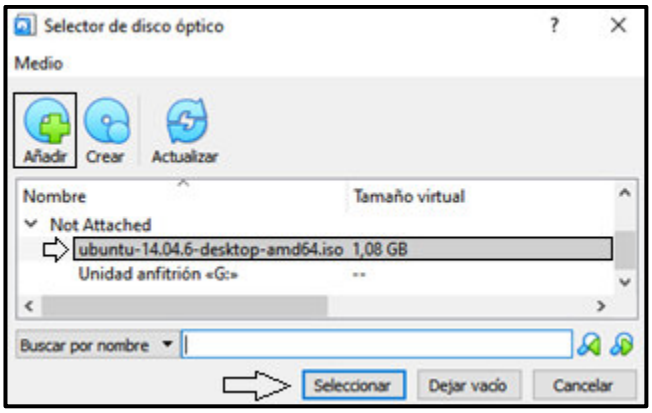


Figura 3.15 Selección de imagen ISO *Ubuntu 14.04.6*

Se debe guardar los cambios realizados y seleccionar la opción “Iniciar” para continuar, ver en la Figura 3.16.

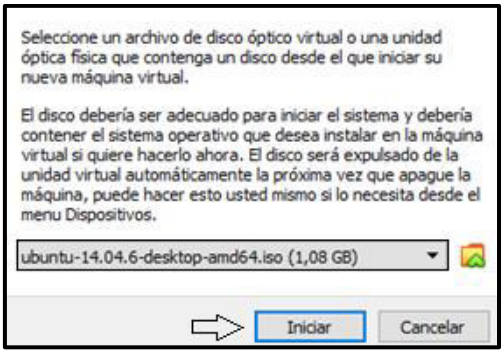


Figura 3.16 Arranque de máquina virtual *Ubuntu*

Se inicia el proceso de instalación y se procede a seleccionar la opción “Instalar *Ubuntu*”, ver en la Figura 3.17.

Figura 3.17 Selección de opción de instalación *Ubuntu*

Se procede a seleccionar “Continuar” para confirmar las características mínimas de instalación, ver en la Figura 3.18.

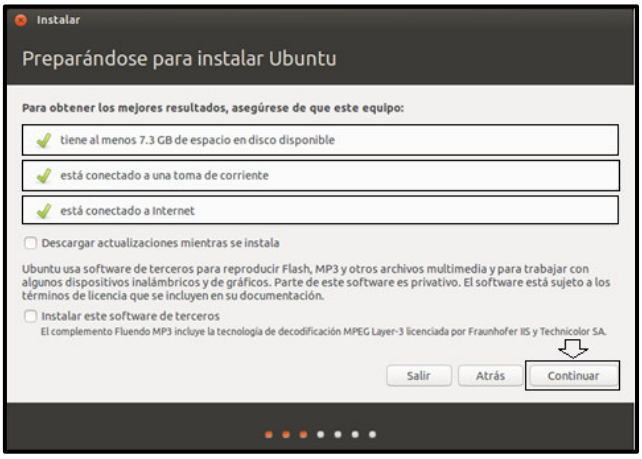


Figura 3.18 Consideraciones mínimas de instalación en *Ubuntu*

Se debe seleccionar la opción “Borrar el disco e instalar *Ubuntu”* y seleccionar la opción de “Instalar ahora” para continuar, ver en la Figura 3.19.

Figura 3.19 Eliminación de disco e instalación de *Ubuntu*

Se debe seleccionar la opción “Continuar” para establecer los cambios del disco y crear las particiones de instalación, ver en la Figura 3.20.

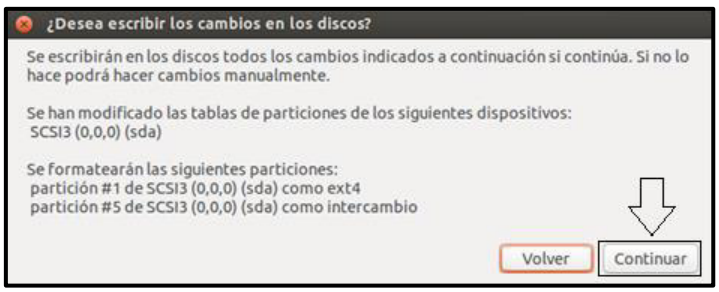


Figura 3.20 Cambio y particiones de disco en instalación de *Ubuntu*

Se debe seleccionar la franja horaria “Guayaquil” y seleccionar la opción “Continuar”, ver en la Figura 3.21.

Figura 3.21 Ubicación geográfica de máquina *Ubuntu*

Se debe seleccionar la opción “español (Latinoamericano*)”* para establecer el idioma por defecto en el teclado y “Continuar” para seguir con la instalación, ver en la Figura 3.22.

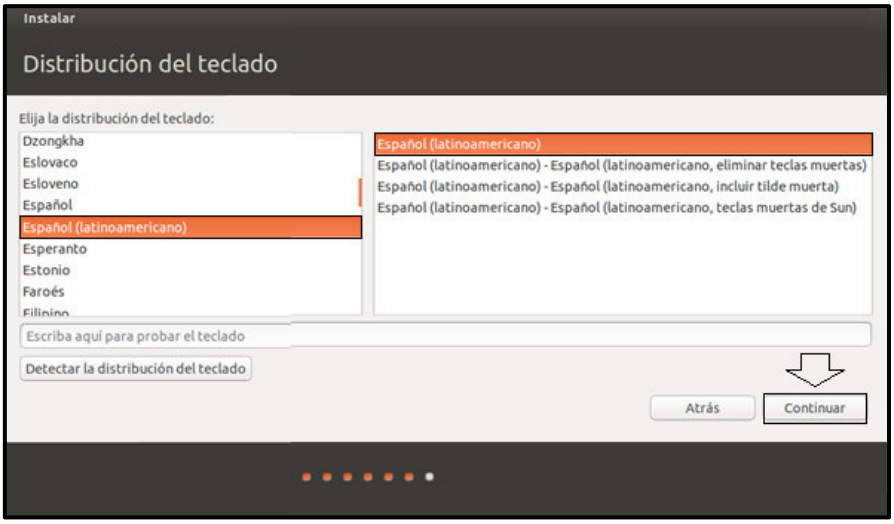


Figura 3.22 Selección de idioma en teclado en *Ubuntu*

Se debe ingresar los datos de establecimiento de la máquina; tales como el nombre del propietario, nombre del equipo, usuario, contraseña e inicio de sesión. Además, se debe configurar una contraseña corta por facilidad académica. Al finalizar se debe seleccionar “Continuar”, ver en la Figura 3.23.

Figura 3.23 Personalización del equipo *Ubuntu*

La instalación de *Ubuntu* se procede a cargar mientras se muestra las diferentes características del equipo, ver en la Figura 3.24.

Figura 3.24 Características de instalación de *Ubuntu*

Se debe seleccionar “Reiniciar Ahora” para continuar, ver en la Figura 3.25.

Figura 3.25 Reinicio del equipo en instalación de *Ubuntu*

Se debe ingresar por teclado la opción ” *Enter*” para inicializar la máquina, ver en la Figura 3.26.

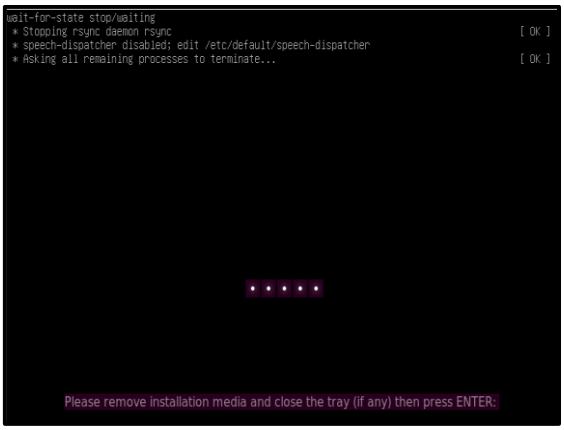


Figura 3.26 Procesamiento de instalación de *Ubuntu*

Se debe ingresar al sistema *Ubuntu* digitando la contraseña, ver en la Figura 3.27.

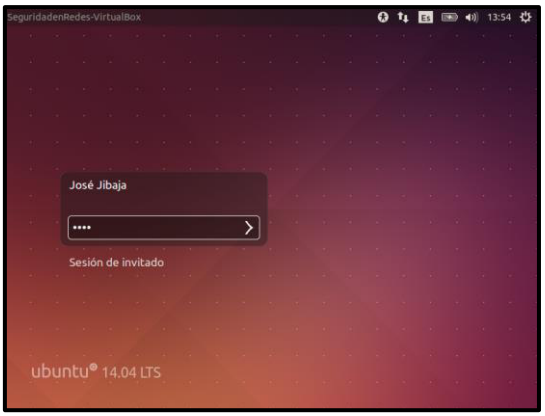


Figura 3.27 Ingreso y validación de credenciales

Se puede observar el escritorio del sistema *Ubuntu* en la finalización de la instalación, ver en la Figura 3.28.

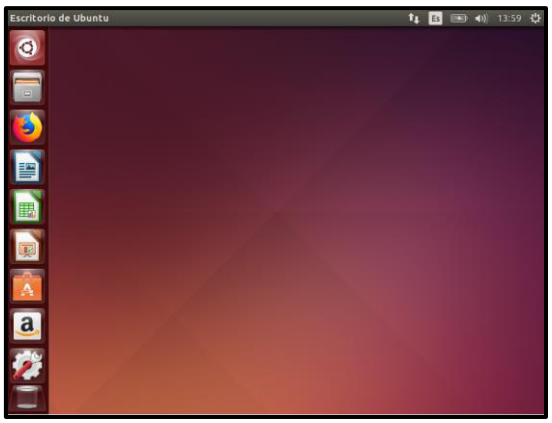
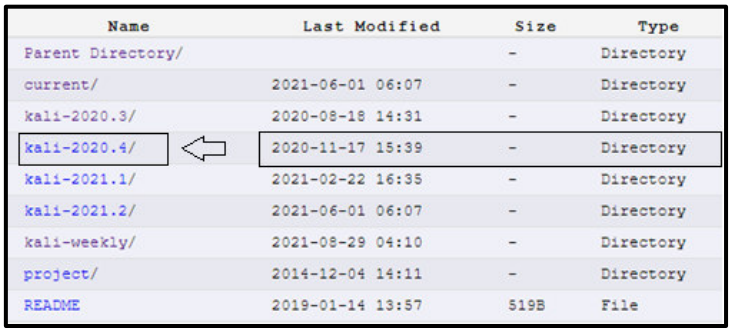


Figura 3.28 Escritorio *Ubuntu*

Descarga del sistema operativo *Kali Linux 2020.4*

Se realiza la descarga de la imagen ISO del sistema operativo *Kali Linux 2020.4*.

La imagen ISO se encuentra disponible desde el repositorio *Mirror* EPN: https://mirror.epn.edu.ec/kali-images/kali-2020.4/. La descarga se realiza mediante la acción "*click*" sobre el nombre de imagen ISO en la columna de "*Name"*, ver en la Figura 3.29.



Name	Last Modified	Size	Type
Parent Directory/		-	Directory
current/	2021-06-01 06:07	-	Directory
kali-2020.3/	2020-08-18 14:31	-	Directory
kali-2020.4/	2020-11-17 15:39	-	Directory
kali-2021.1/	2021-02-22 16:35	-	Directory
kali-2021.2/	2021-06-01 06:07	-	Directory
kali-weekly/	2021-08-29 04:10	-	Directory
project/	2014-12-04 14:11	-	Directory
README	2019-01-14 13:57	519B	File

Figura 3.29 Descarga de imagen ISO de *Kali Linux 2020.4*

Creación de *Kali Linux* en *Oracle VM VirtualBox*

Una vez descargada la imagen ISO de *Kali Linux 2020.4*, se procede a crear una máquina virtual para el sistema operativo.

Se debe ingresar a la herramienta *Oracle VM VirtualBox* y posteriormente seleccionar la opción "Nueva" para creación de máquina virtual, ver en la Figura 3.30.

Figura 3.30 Creación de máquina virtual *Kali Linux*

A continuación, se debe ingresar las características del equipo tales como el nombre, tipo de sistema y versión de trabajo. Además, se debe seleccionar la opción "*Next*" para continuar, ver en la Figura 3.31.

Figura 3.31 Características de máquina virtual *Kali Linux*

Se asigna el tamaño de la memoria RAM a un valor de un 1 (GB) como requerimiento básico del módulo y selecciona la opción "*Next*" para continuar, ver en la Figura 3.32.

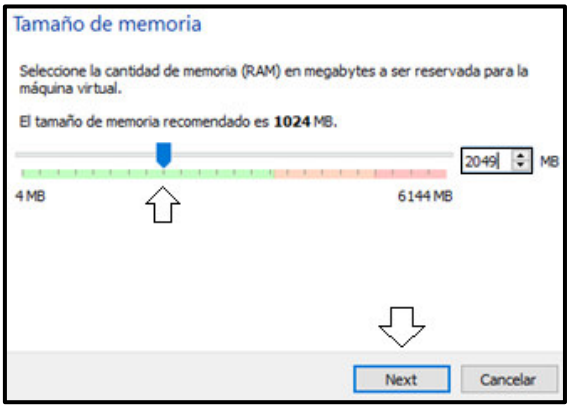


Figura 3.32 Asignación de memoria RAM en *Kali Linux*

Se debe seleccionar la opción "Crear un disco duro virtual" y la opción "Crear" para continuar, ver en la Figura 3.33.

Figura 3.33 Configuración de disco duro virtual en *Kali Linux*

Se debe seleccionar la opción de "Reservado dinámicamente" y seleccionar "*Next*" para continuar, ver en la Figura 3.34.

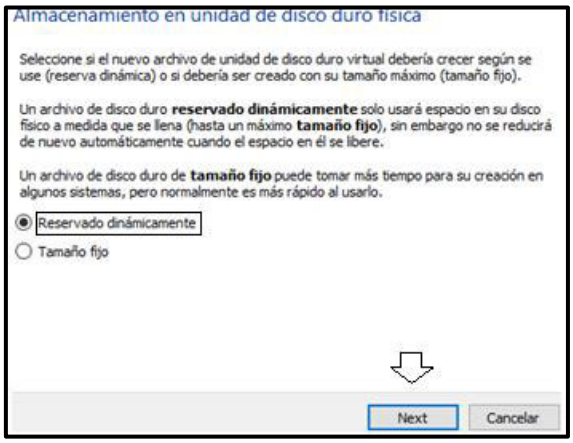


Figura 3.34 Adjudicación de almacenamiento dinámico en *Kali Linux*

Se muestra la ruta de ubicación de la máquina y el tamaño del disco duro. Se debe seleccionar la opción “Crear” para continuar, ver en la Figura 3.35.

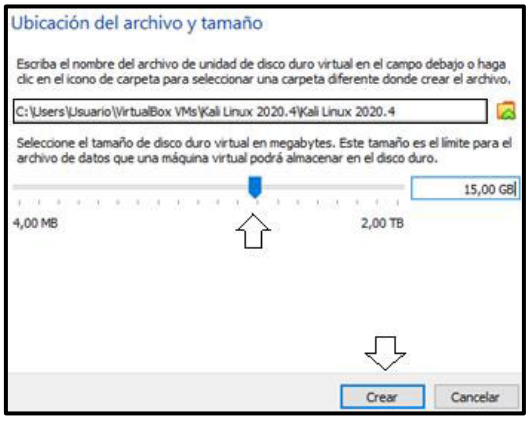


Figura 3.35 Ruta de ubicación de máquina virtual *Kali Linux*

Instalación *Kali Linux 2020.4*

Se ingresa a la opción de “Configuración”, en el apartado de “Sistema” e ingresa a “Procesador”, para aumentar el procesador de la máquina a 2 CPU, ver en la Figura 3.36.

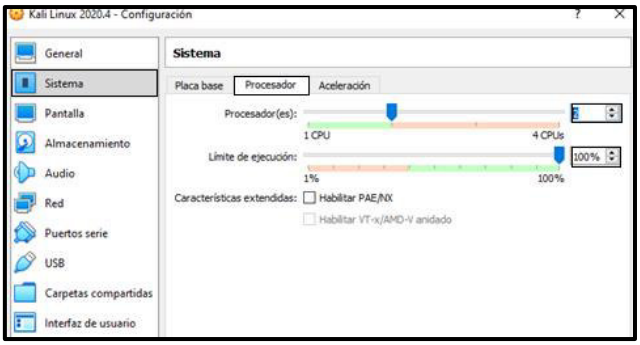


Figura 3.36 Configuración de procesador virtual en *Kali Linux*

Se debe dirigir a la opción de “Pantalla” en el apartado de “Aceleración” y habilitar la aceleración 3D, ver en la Figura 3.37.

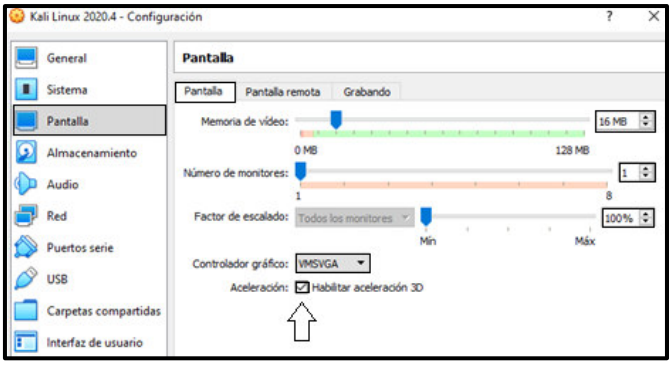


Figura 3.37 Configuración de pantalla virtual en *Kali Linux*

Se ingresa a la carpeta de almacenamiento de imágenes ISO, ver en la Figura 3.38.

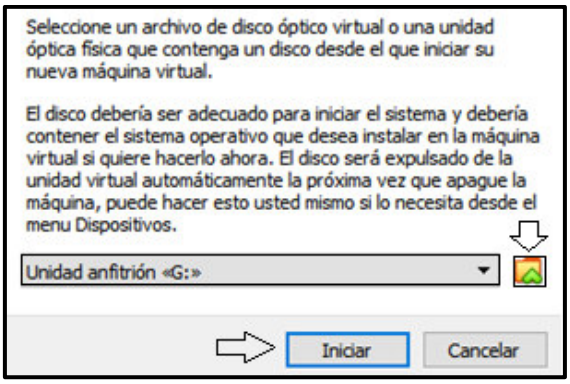


Figura 3.38 Carpeta de imágenes ISO

Se debe seleccionar la imagen ISO del sistema *Kali Linux*, ver en la Figura 3.39.

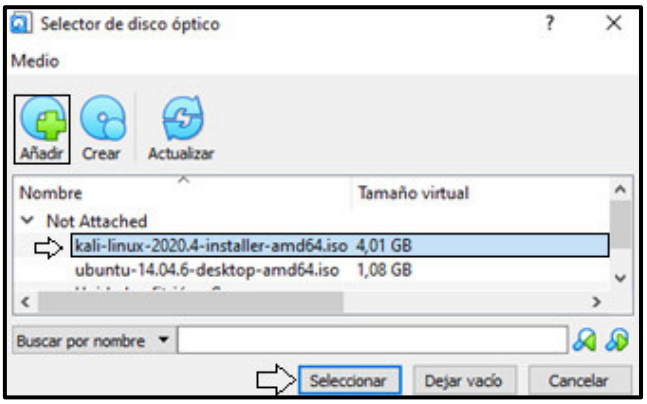


Figura 3.39 Selección de imagen ISO *Kali Linux 2020.4*

Se debe guardar los cambios realizados y seleccionar la opción “Iniciar” para continuar, ver en la Figura 3.40.

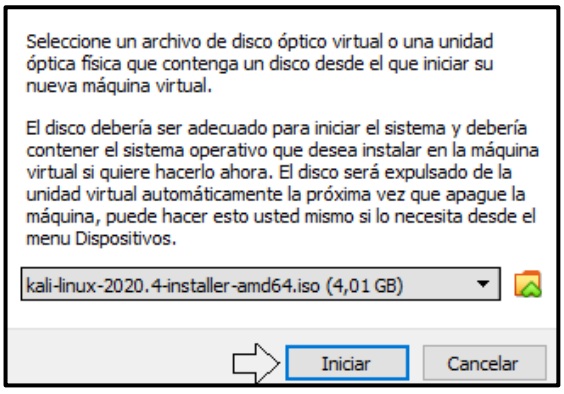


Figura 3.40 Arranque de máquina virtual *Kali Linux*

Se inicia el proceso de instalación y se procede a seleccionar la opción “*Graphical install*”, ver en la Figura 3.41.

Figura 3.41 Instalación en modo grafico en *Kali Linux*

De igual forma se procede a seleccionar el idioma por defecto de instalación en la opción “español”, ver en la Figura 3.42.

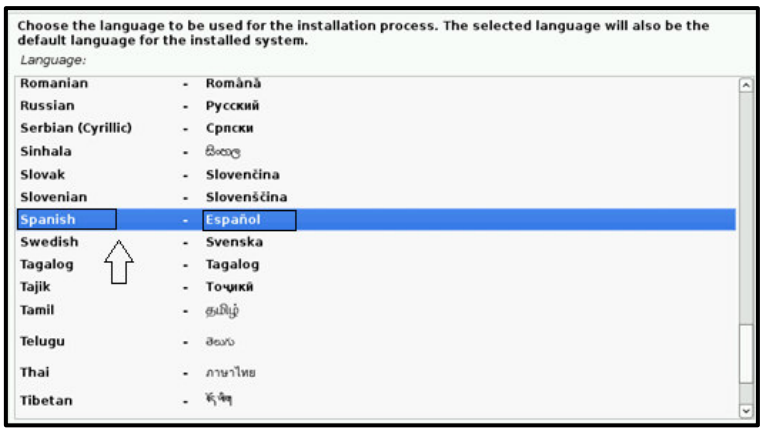


Figura 3.42 Selección del idioma en *Kali Linux*

Se configura la ubicación geográfica del equipo en la opción “Ecuador”, ver en la Figura 3.43.

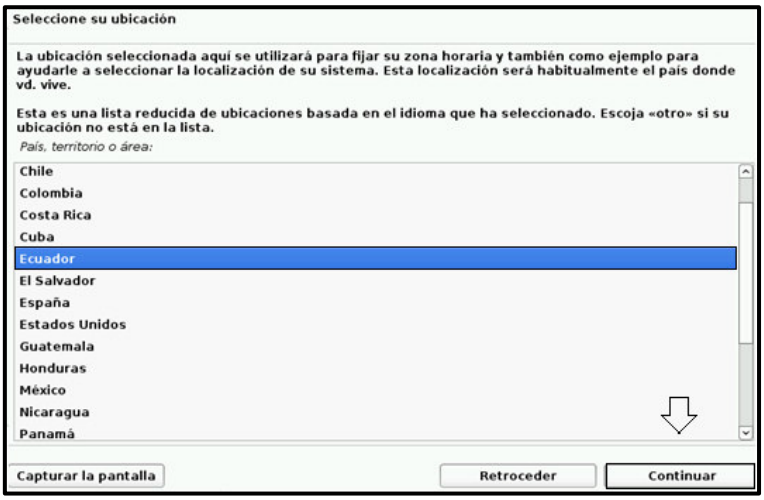


Figura 3.43 Ubicación geográfica del equipo *Kali Linux*

De igual forma se procede a seleccionar el idioma por defecto del teclado en la opción “Latinoamericano”, ver en la Figura 3.44.

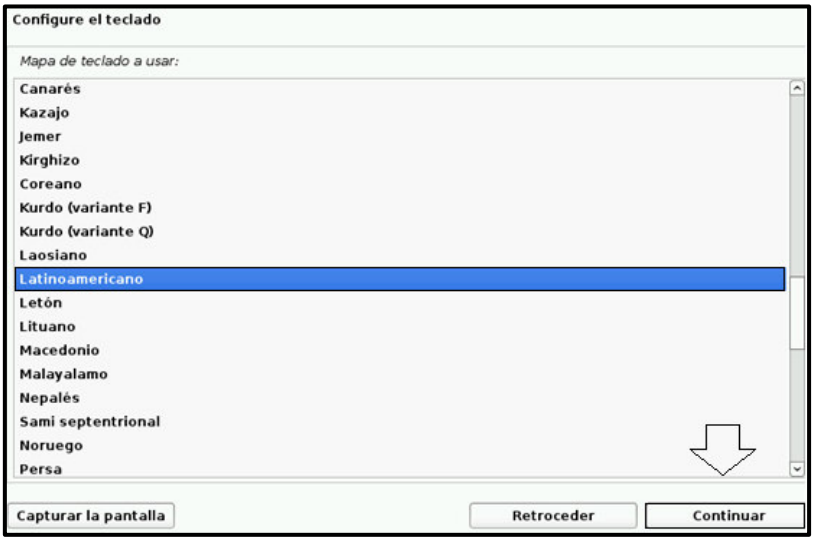


Figura 3.44 Configuración del idioma en el teclado en *Kali Linux*

De igual forma se detalla el procesamiento de los componentes de la instalación media, ver en la Figura 3.45.

Figura 3.45 Procesamiento de componentes adicionales

A continuación, se procede a configurar el nombre del equipo, ver en la Figura 3.46.

Figura 3.46 Establecimiento del nombre del equipo “*Kali*”

Se debe considerar el nombre del dominio, si no es así el caso, se procede a seleccionar la opción “Continuar”, ver en la Figura 3.47.

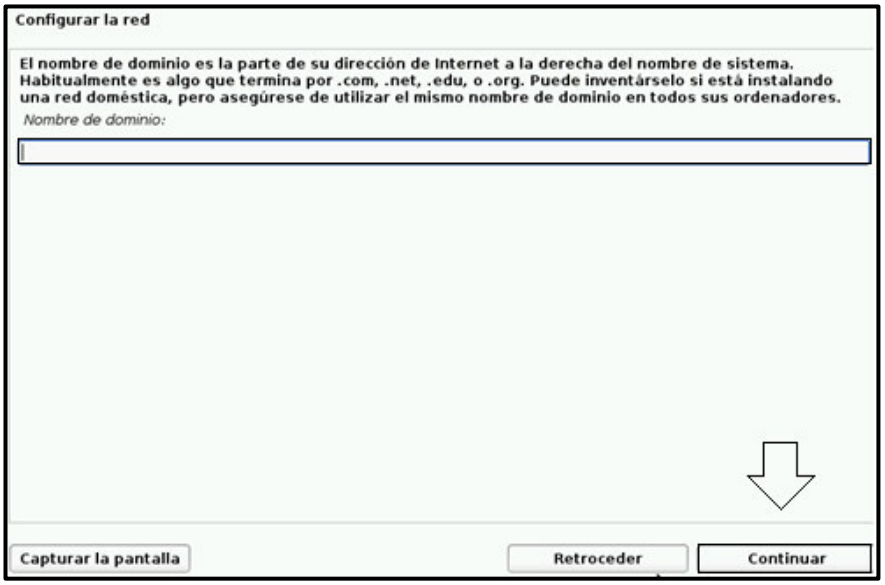


Figura 3.47 Ingreso de nombre de domino en *Kali Linux*

Se establece el nombre del equipo y se selecciona la opción “Continuar”, ver en la Figura 3.48.

Figura 3.48 Creación de cuenta de usuario en *Kali Linux*

Se configura la contraseña de la cuenta del usuario con una serie de caracteres alfabéticos y numéricos, ver en la Figura 3.49.

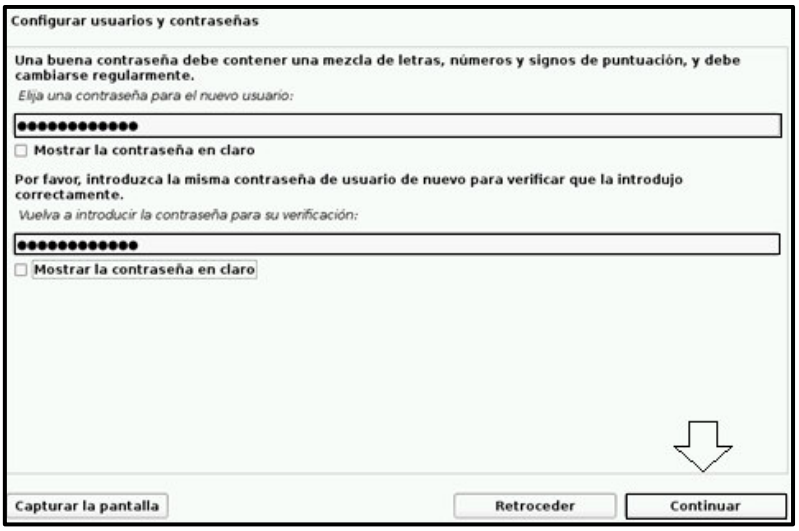


Figura 3.49 Asignación de contraseña en *Kali Linux*

Se selecciona la localización del equipo previamente configurada, ver en la Figura 3.50.

Figura 3.50 Asignación de zona horaria en *Kali Linux*

De igual forma se detalla el procesamiento de las particiones del disco en la instalación, ver en la Figura 3.51.

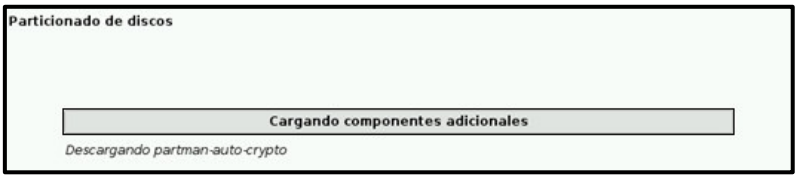


Figura 3.51 Particiones de discos en carga en *Kali Linux*

Se debe elegir la opción de “Guiado-utilizar todo el disco”, con el fin de seleccionar todo el almacenamiento en el disco, ver en la Figura 3.52.

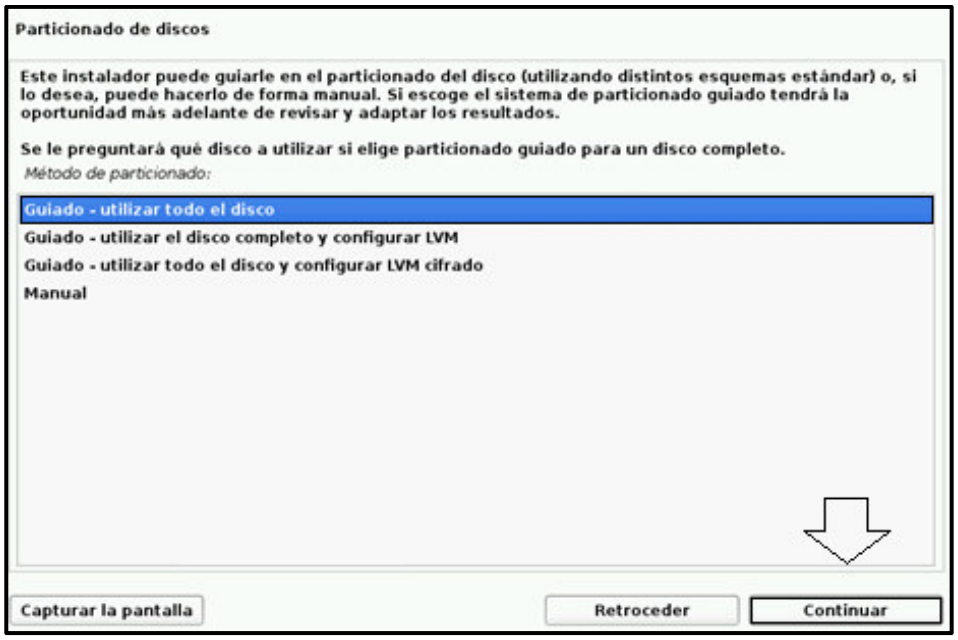


Figura 3.52 Guía de establecimiento de partición de disco en *Kali Linux*

A continuación, se optar por la opción “Continuar” para el establecimiento de la limpieza o formateo del disco, ver en la Figura 3.53.

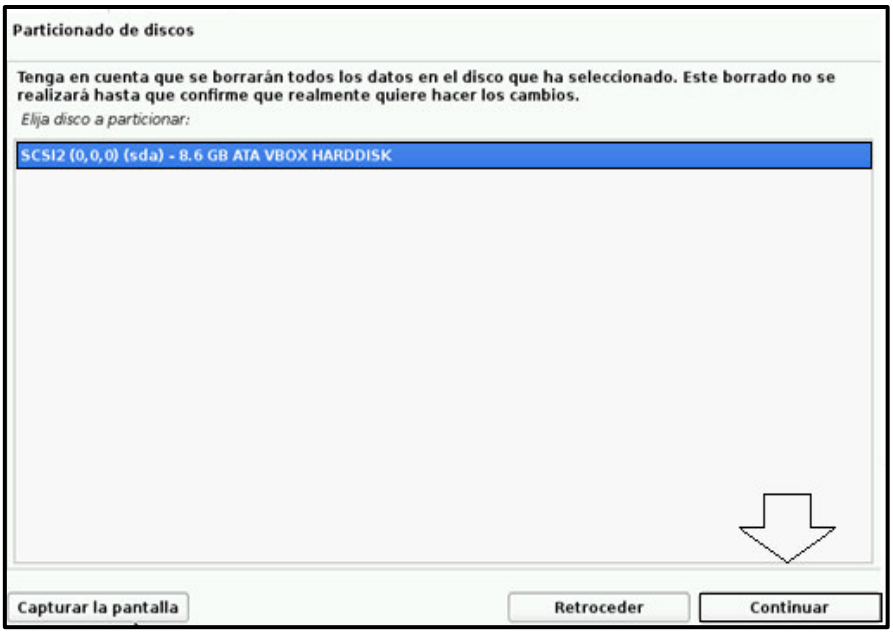


Figura 3.53 Inicialización de disco limpio en *Kali Linux*

Del mismo modo, se debe elegir por la opción “Continuar” para establecer ficheros en una sola partición, ver en la Figura 3.54.

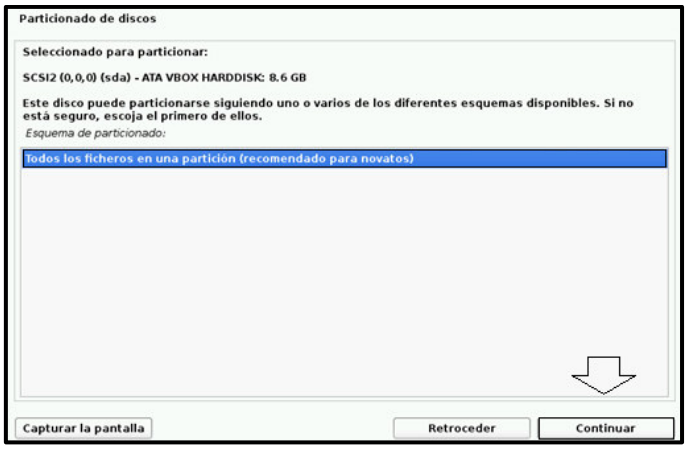


Figura 3.54 Configuración de partición de disco en *Kali Linux*

De igual forma se procede a escoger la opción “Continuar” para establecer las configuraciones, ver en la Figura 3.55.

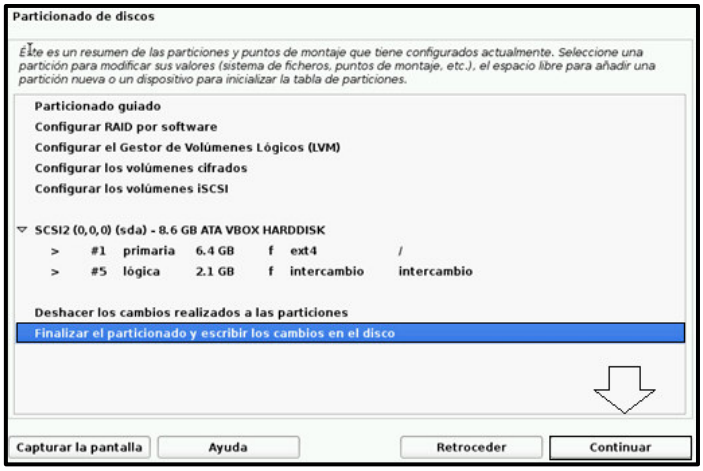


Figura 3.55 Finalización de partición de disco en *Kali Linux*

Se procede a optar por la opción “sí” para establecer las particiones de la instalación y seleccionar la opción “Continuar”, ver en la Figura 3.56.

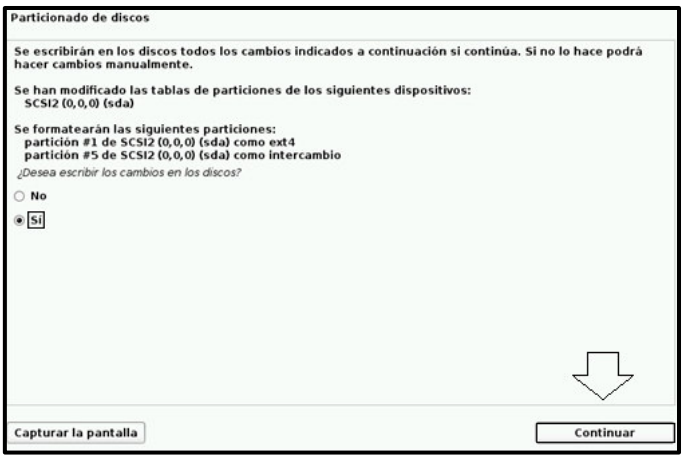


Figura 3.56 Confirmación de particiones en *Kali Linux*

A continuación, se detalla el procesamiento de la instalación del sistema base, ver en la Figura 3.57.

Figura 3.57 Procesamiento de instalación del sistema base en *Kali Linux*

Se debe elegir la opción “Continuar” debido a que no se realiza la opción de configuración de un servidor *Proxy*, ver en la Figura 3.58.

Figura 3.58 Configuración de *Proxy* no efectuada en *Kali Linux*

Del mismo modo, se detalla el procesamiento de la configuración del gestor de paquetes, ver en la Figura 3.59.

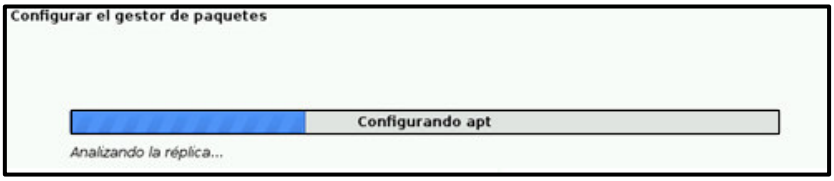


Figura 3.59 Procesamiento de configuración de gestor de paquetes en *Kali Linux*

A continuación, se procede a seleccionar los programas por defecto en la distribución, ver en la Figura 3.60.

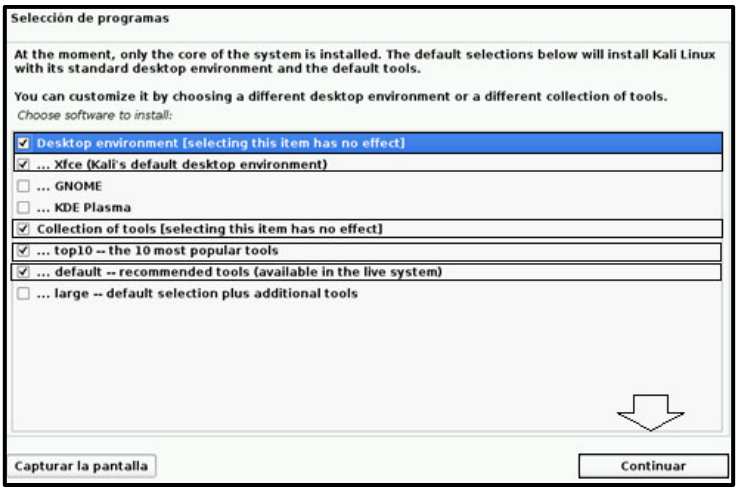


Figura 3.60 Configuración de programas y herramientas en *Kali Linux*

Del mismo modo, se detalla el procesamiento de la instalación de los programas y herramientas en *Kali Linux*, ver en la Figura 3.61.

Figura 3.61 Procesamiento de herramientas en *Kali Linux*

Se procede a escoger la opción “Sí” y “Continuar” para el arranque principal del sistema mediante GRUB, ver en la Figura 3.62.

Figura 3.62 Configuración de entorno gráfico GRUB en *Kali Linux*

A continuación, se debe optar por la opción “Continuar” para establecer el inicio de arranque de la herramienta *GRUB*, ver en la Figura 3.63.

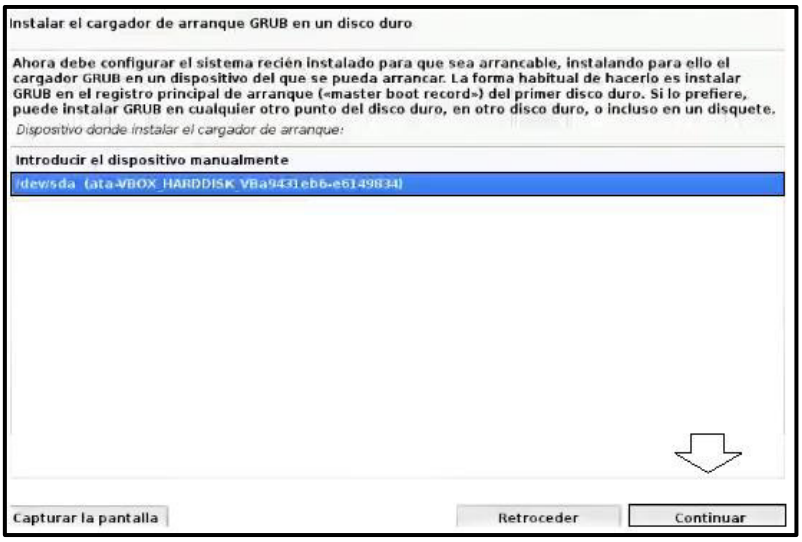


Figura 3.63 Instalación de GRUB en *Kali Linux*

Para finalizar el proceso de instalación, se debe elegir la opción “Continuar”, ver en la Figura 3.64.

Figura 3.64 Finalización de instalación de máquina *Kali Linux*

3.2 Instalación de software de *Honeyd* y manejo de sistemas operativos

Análisis de librerías de *Honeyd*

La estructura de instalación de *Honeyd* se sustenta en el uso de librerías enfocadas en el manejo de paquetes y creación de archivos. Además, comprende de herramientas complementarias de corrección de errores.

A continuación, se presenta las librerías más predominantes:

- ***Libevent*:** Es la librería encargada de la notificación de eventos. Es un mecanismo de señales de tiempo regulares para la devolución de llamadas de un descriptor de archivos [32].
- ***Libdnet***: Es la librería encargada de la creación de paquetes. Se enfoca en proporcionar una interfaz portátil y simplificada para diferenciar las rutinas de la red de bajo nivel. Se dispone de la configuración de tablas de ARP, configuraciones de interfaz, *firewalling*, configuraciones de direcciones de red, tunelización IP y transmisión de datagramas [32].
- ***Libpcap***: Es la librería encargada del rastreo de paquetes. Se configura la interfaz independiente al sistema, enfocando la captura de paquetes a nivel de usuario. Proporciona un monitoreo de red de bajo nivel en base a un marco portátil. El manejo de las aplicaciones incluye la depuración de la red, seguimiento de seguridad, recolección de estadística de red entre otras [32].

Desarrollo de instalación de *Honeyd*

La instalación de la herramienta *Honeyd* se desarrolla en el sistema operativo *Ubuntu*, por lo cual se realiza los siguientes pasos:

Descargas de librerías

Se requiere el ingreso al sistema operativo *Ubuntu* y se procede a abrir la terminal e ingresar al usuario administrador "*root*" mediante el comando "*sudo*" y la opción "*su*".

$ sudo su

Se opta en descargar las dependencias y librerías por medio del comando "*wget*" con la dirección web del archivo.

 # wget "dirección web"

Listado de descargas

El proceso de descarga de librerías y herramientas se realiza por medio de las direcciones web gestionadas por el comando "*wget*".
Las librerías y documentos se describen en la Tabla 3.6.

Tabla 3.6 Archivos de instalación *Honeyd* [33] [34] [35] [36] [37] [38] [39]

Archivo	Paquete	Página Web
Libevent	libevent-1.4.14b-stable	https://monkey.org/~provos/libevent-1.4.14b-stable.tar.gz
Libdnet	libdnet-1.11	http://jaist.dl.sourceforge.net/sourceforge/libdnet/libdnet-1.11.tar.gz
Libpcap	libpcap-1.3.0	http://repository.timesys.com/buildsources/l/libpcap/libpcap-1.3.0/libpcap-1.3.0.tar.gz
Honeyd 1.5c	Honeyd-1.5c	http://www.Honeyd.org/uploads/Honeyd-1.5c.tar.gz
Zlib	zlib-1.2.8	https://zlib.net/fossils/zlib-1.2.8.tar.gz
Arpd	arpd-0.2	http://download.openpkg.org/components/cache/arpd/aprd-0.2.tar.gz
Honeyd Kit	Honeyd_kit-1.0c-a	http://www.citi.umich.edu/u/provos/Honeyd/Honeyd_kit-1.0c-a.tgz

El empleo del comando "*ls*" permite visualizar los archivos descargados en el directorio principal del usuario, ver en la Figura 3.65.

Figura 3.65 Descarga de archivos de instalación de *Honeyd*

Creación del directorio "*Honeyd*"

El directorio "*Honeyd*" se crea mediante la utilización del comando "*mkdir*", ver en la Figura 3.66.

Figura 3.66 Creación del directorio "*Honeyd*"

Cambio de directorio

El traslado de los archivos de descarga alojados en el directorio “Usuario” hacia el directorio “*Honeyd*” se realiza mediante el comando “mv”. Además, el ingreso al directorio y la visualización de los archivos se realiza mediante el comando “*ls*”, ver en la Figura 3.67.

Figura 3.67 Almacenamiento de archivos en directorio *“Honeyd”*

Instalación de Compilador de GNU “*g++ gcc*”

Se opta por instalar el paquete Compilador de *GNU* en lenguaje *C++* como requisito de instalación de *Honeyd*, ver en la Figura 3.68.

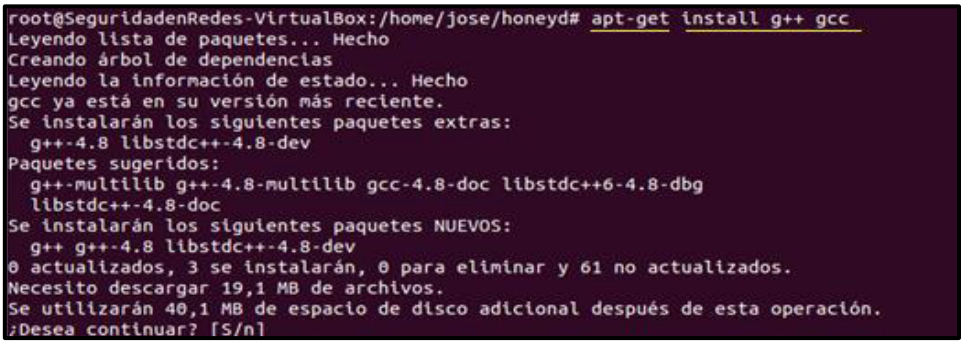


Figura 3.68 Instalación de Compilador *GNU*

Descompresión e Instalación de Archivos

Libevent

Se debe descomprimir el archivo *Libevent*, ver en la Figura 3.69.

- *# tar -zxvf libevent-1.4.14b-stable.tar.gz*

Figura 3.69 Descompresión de archivo *Libevent*

De igual forma se opta por ejecutar los comandos de instalación descritos de la siguiente manera:

- *# cd libevent-1.4.14b-stable*
- *# ./configure*
- *# make*
- *# make install*

Libdnet

Se debe descomprimir el archivo *Libdne*t, ver en la Figura 3.70.

- *# tar -zxvf libdnet-1.11.tar.gz*

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd# tar -zxvf libdnet-1.11.tar.gz
libdnet-1.11/
libdnet-1.11/acconfig.h
libdnet-1.11/aclocal.m4
libdnet-1.11/config/
libdnet-1.11/config/acinclude.m4
libdnet-1.11/config/config.guess
libdnet-1.11/config/config.sub
libdnet-1.11/config/install-sh
libdnet-1.11/config/ltmain.sh
```

Figura 3.70 Descompresión de archivo *Libdnet*

De igual forma se opta por ejecutar los comandos de instalación descritos de la siguiente manera:

- *# cd libdnet-1.11*
- *# ./configure*
- *# make*
- *# make install*

Libpcap

Se debe descomprimir el archivo *Libpcap*, ver en la Figura 3.71.

- *# tar -zxvf libpcap-1.3.0.tar.gz*

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd# tar -zxvf libpcap-1.3.0.tar.gz
libpcap-1.3.0/
libpcap-1.3.0/sf-pcap-ng.h
libpcap-1.3.0/pcap-dlpi.c
libpcap-1.3.0/pcap_list_tstamp_types.3pcap.in
libpcap-1.3.0/pcap-dag.c
libpcap-1.3.0/pcap-bpf.c
```

Figura 3.71 Descompresión de archivo *Libpcap*

De igual forma se opta por ejecutar los comandos de instalación descritos de la siguiente manera:

- *# cd libpcap-1.3.0*
- *# ./configure*

Se observa la solicitud de la instalación del paquete "*flex*", ver en la Figura 3.72.

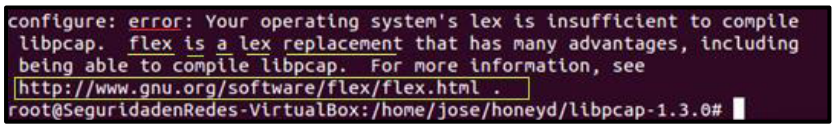


Figura 3.72 Solicitud de descarga de paquete *flex*

Se debe descarga e instalar el paquete "*flex*", ver en la Figura 3.73.

- *# sudo apt-get install flex*

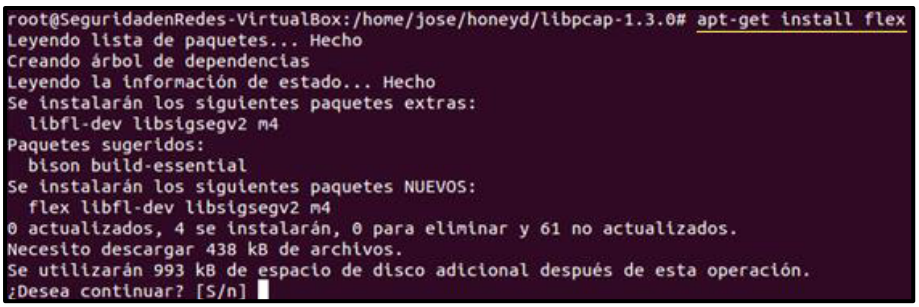


Figura 3.73 Instalación de paquete *flex*

De igual forma se continúa el proceso de instalación con los siguientes comandos:

- *# ./configure*
- *# make*

Se observa la solicitud de la instalación del paquete gramatical, ver en la Figura 3.74.

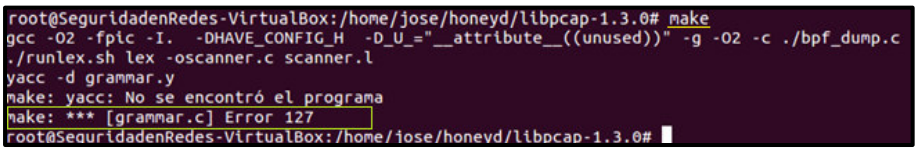


Figura 3.74 Solicitud de descarga de paquete *bison*

El paquete “flex” se acompaña de un paquete gramatical denominado “*bison*”, se opta por instalar el paquete “*bison*”, ver en la Figura 3.75.

- *# sudo apt-get install bison*

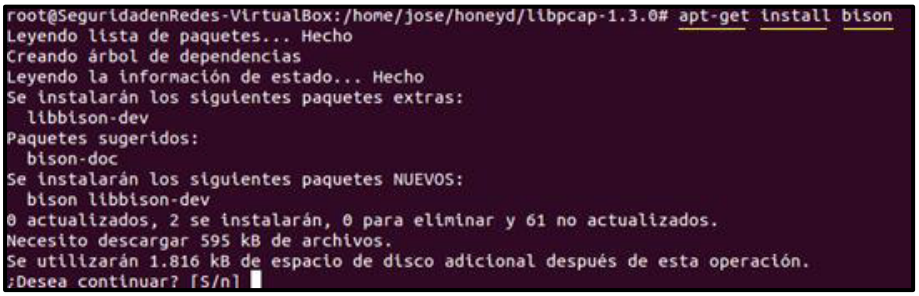


Figura 3.75 Instalación de paquete *bison*

De igual forma finaliza el proceso mediante el empleo de los siguientes comandos :

- *# make*
- *# make install*

Honeyd **1.5c**

Se debe descomprimir el archivo *Honeyd* 1.5c, ver en la Figura 3.76.

- *# tar -zxvf Honeyd-1.5c.tar.gz*

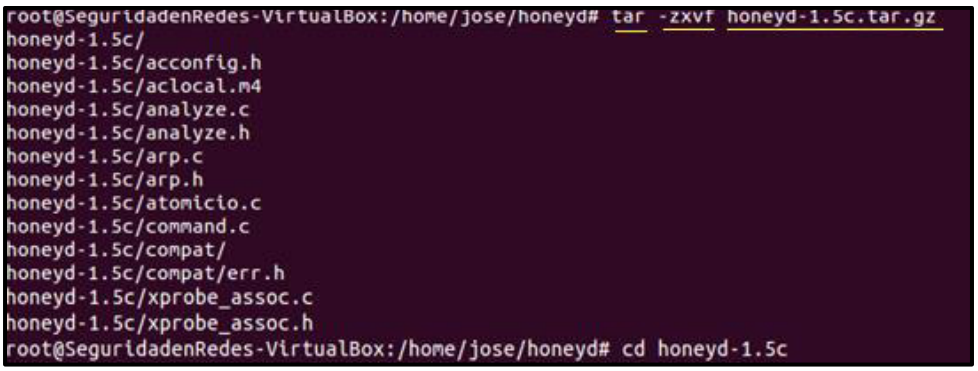


Figura 3.76 Descompresión de archivo *Honeyd* 1.5c

De igual forma se opta por ejecutar los comandos de instalación descritos de la siguiente manera:

- *# cd Honeyd-1.5c*
- *# ./configure*

Se observa la solicitud de la instalación del paquete "*Libedit*", ver en la Figura 3.77.

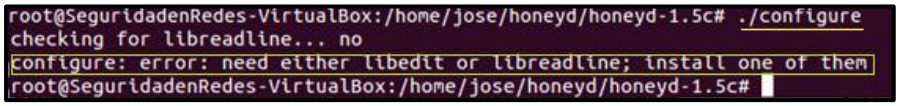


Figura 3.77 Solicitud de descarga de paquete *Libedit*

Se debe instalar el paquete "*Libedit*", ver en la Figura 3.78.

- *# sudo apt-get install libedit-dev*

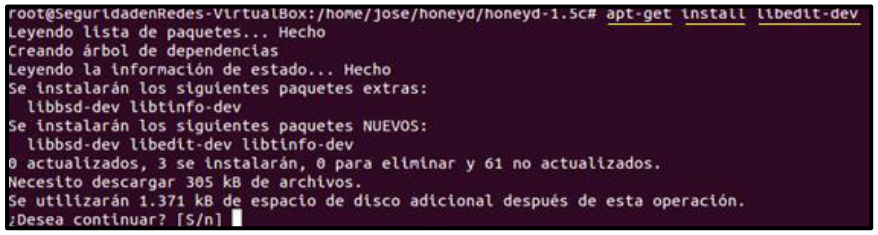


Figura 3.78 Instalación de paquete *Libedit*

- *# ./configure*

Se observa la solicitud de la instalación del paquete "*Zlib"*, ver en la Figura 3.79.

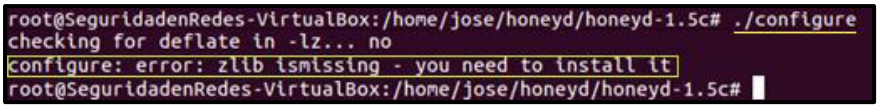


Figura 3.79 Solicitud de descarga de paquete *Zlib*

Se opta por ingresar al directorio principal mediante el comando "cd" e ingresar al directorio del paquete descargado *"Zlib"*.

Zlib

Se debe descomprimir el archivo *Zlib*, ver en la Figura 3.80.

- *# tar -zxvf zlib-1.2.8.zip*

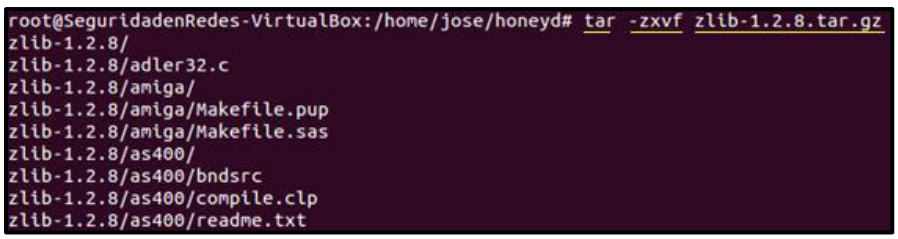


Figura 3.80 Descompresión de archivo *Zlib*

De igual forma se continúa el proceso mediante la ejecución de los comandos de instalación descritos de la siguiente manera:

- *# cd zlib-1.2.8*
- *# ./configure*
- *# make*
- *#make install*

Se ejecuta los comandos de continuidad de *Honeyd* 1.5c:

- *# cd ../Honeyd-1.5c*
- *#./configure*
- *# ln -s /lib/x86_64-linux-gnu/libc.so.6 /usr/lib/libc.so*
- *# ./ configure*

Se recompila el archivo *Lipcap*-1.3.0, mediante los siguientes comandos**:**

- *# cd ../libpcap-1.3.0*
- *# make clear*
- *#./configure*
- *# make*
- *# make install*

Se ejecuta los comandos de continuidad de *Honeyd* 1.5c:

- *# cd Honeyd-1.5c*
- *# ./configure*
- *# make*
- *# make install*

Arpd

Se debe descomprimir el archivo *Arpd*, ver en la Figura 3.81.

- *# tar -zxvf arpd-0.2.tar.gz*

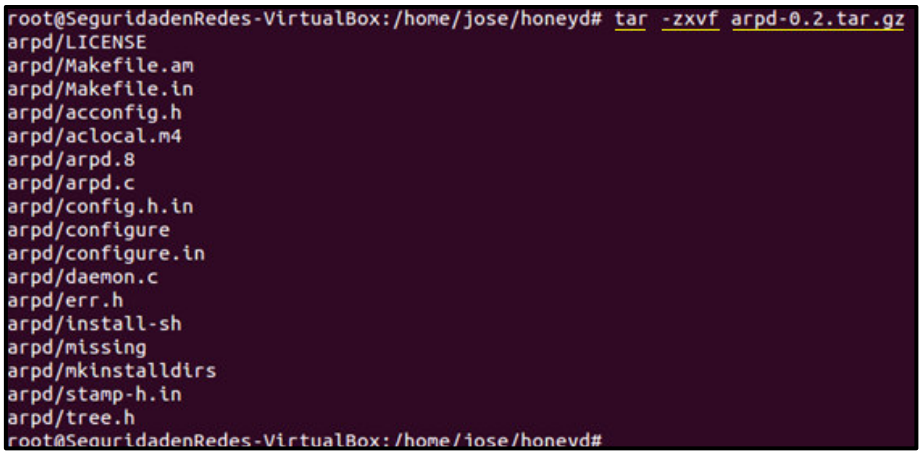


Figura 3.81 Descompresión de archivo *Arpd*

De igual forma se opta por ejecutar los comandos de instalación descritos de la siguiente manera:

- *# cd arpd*
- *# ./configure*

Se agrega la opción "*#define __FUNCTION__* "en el archivo "*arpd.c*, ver en la Figura 3.82.

- *# vim arpd.c*

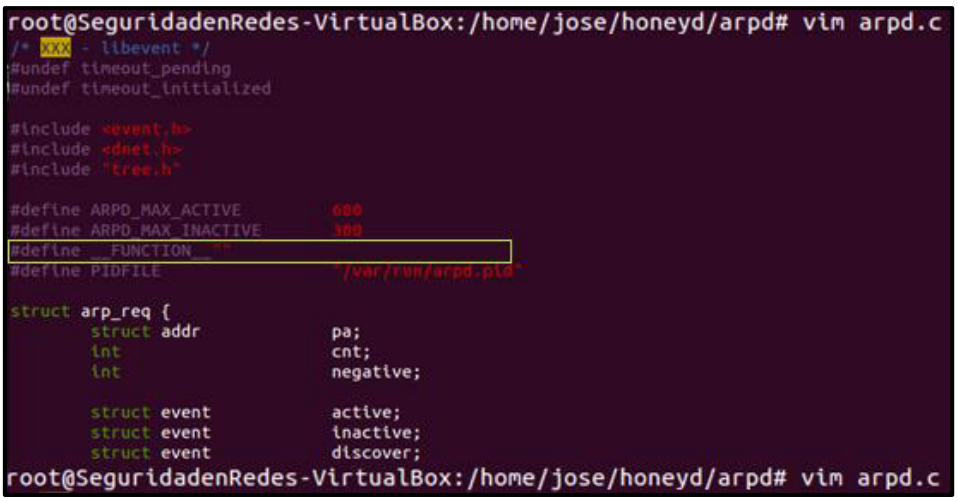


Figura 3.82 Establecimiento de función en archivo "*arpd.c*"

- *# make*
- *# make install*

Honeyd Kit

Se debe descomprimir el archivo *Honeyd Kit*, ver en la Figura 3.83.

- *# tar -zxvf Honeyd_kit-1.0c-a.tgz*

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd# tar -zxvf honeyd_kit-1.0c-a.
tgz
honeyd_kit-1.0c-a/
honeyd_kit-1.0c-a/arpd
honeyd_kit-1.0c-a/docs/
honeyd_kit-1.0c-a/docs/citi-tr-03-1.pdf
honeyd_kit-1.0c-a/docs/INSTALL.kuang2
honeyd_kit-1.0c-a/docs/README.cmdexe
honeyd_kit-1.0c-a/docs/README.kuang2
honeyd_kit-1.0c-a/docs/README.mydoom
honeyd_kit-1.0c-a/docs/simulating_networks_with_honeyd.pdf
honeyd_kit-1.0c-a/honeyd
honeyd_kit-1.0c-a/honeyd.conf
```

Figura 3.83 Descompresión de archivo *Honeyd Kit*

De igual forma se opta por ejecutar los comandos de instalación descritos de la siguiente manera:

- *# cd Honeyd_kit-1.0c-a*
- *# ./configure*
- *# make*
- *# make install*

Archivos y configuración del sistema

Archivos *Honeyd*

Honeyd se compone de varios elementos para la creación y funcionamiento de los *Honeypots*. Mediante el comando "*ls"* en el directorio "*Honeyd_kit-1.0c-a*" se visualizan los archivos, ver en la Figura 3.84.

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd/honeyd_kit-1.0c-a# ls
arpd          honeyd.conf.bloat     logs          README           xprobe2.conf
docs          honeyd.conf.networks  nmap.assoc    scripts
honeyd        honeyd.conf.simple    nmap.prints   start-arpd.sh
honeyd.conf   honeyd.org            pf.os         start-honeyd.sh
```

Figura 3.84 Archivos de *Honeyd*

A continuación, se describe de manera general los archivos de *Honeyd*:

- ***Arpd.-*** Archivo de condición binaria o *script* para el establecimiento de conexión de direcciones IP por medio del protocolo ARP [28].
- ***Docs.-*** Documentación proporcionada y desarrollada por la comunidad de *Honeyd* [28].

- ***Honeyd*.** Archivo de condición binaria o *script* para la interacción y detección de atacantes en la acción de comando [28].
- ***Honeyd.conf.-*** Archivo de configuración *Honeyd* [28].
- ***Honeyd.conf.simple***.- Archivo de configuración simple de *Honeyd* para manejo de pruebas de funcionamiento [28].
- ***Honeyd.conf.bloat***.- Archivo de configuración avanzado de *Honeyd* para manejo de capacidades amplias de *Honeypots* [28].
- ***Honeyd.conf.networks.-*** Archivo de configuración avanzado de *Honeyd* para manejo de capacidades amplias de red [28].
- ***Honeyd.org.-*** Archivo de *Honeyd* [28].
- ***Logs.-*** Directorio opcional para registros adicionales [28].
- ***Nmap.assoc.-*** Archivo de *Nmap* asociado con base de datos de huellas digitales específicas de sistemas operativos generales en función de personalidades [28].
- ***Nmap.prints.-*** Archivo de *Nmap* configurado con base de datos de huellas dactilares de *Honeyd* para establecimiento de *Honeypots* [28].
- ***Pf.os.-*** Archivo de base de datos para huellas digitales pasivas [28].
- ***Xprobe2.conf.-*** Archivo de base de datos para huellas dactilares del sistema operativo conforme a la sonda "*Xprobe2*" constituida en *Honeyd* [28].
- ***README.-*** Información de *Honeyd* [28].
- ***Scripts.-*** *Scripts* de simulación de servicios utilizados por *Honeyd* [28]
- ***Start-arpd.sh.-*** S*cript* de establecimiento de proceso ARP, asignación de red y direcciones IP para manejo de *Honeypots* por medio de *Honeyd* [28].
- ***Start-Honeyd.sh.-*** *Script* de establecimiento de proceso y comando de inicialización de proceso de *Honeyd* [28].

Configuraciones iniciales de *Honeyd*

Cambio de permisos de Usuario

Una vez instalado el demonio de *Honeyd*, se debe conceder permisos de usuario mediante el comando "*chown*" a la condición "*nobody*". La acción permite que *Honeyd* escriba la información y el proceso se ejecute sin error de configuración [28].

- *# chown -R nobody Honeyd-kit-1.0c-a*

Archivos de Registro de Actividad

Se debe crear el archivo de registro de información de *Honeyd* en el directorio "*/var/log*". El proceso se realiza mediante el comando "*mkdir*", con la ruta del directorio y el nombre del archivo asignado. Asimismo, se procede a asignar los permisos de usuario en el directorio [28].

- *# mkdir /var/log/Honeyd*
- *# chown -R nobody /var/log/Honeyd*

Se debe crear el archivo texto para el manejo de archivos de simulación "*scripts*" en el directorio "*var/log*". El proceso se realiza mediante el comando "*touch*", con la ruta del directorio y el nombre del archivo "*Honeyd.txt*". Además, se debe proporcionar los permisos de usuario en el archivo [28].

- *# touch /var/log/Honeyd.txt*
- *# chown -R nobody /var/log/Honeyd.txt*

Pruebas de funcionamiento de *Honeyd*

Inicialización *arpd*

Se opta por inicializar el demonio de *arpd*, el proceso permite proporcionar a la tarjeta de red del sistema *Ubuntu* "*eth0*" un puerto de salida a la Red *Honeypot* 192.168.1.0/24 del módulo virtual, en base a la dirección MAC 08:00:27:ac:20:50 del equipo. El servicio permite la intercomunicación de los equipos *Honeypots* con la red externa y el equipo *Kali Linux*. La acción se realiza mediante el comando "*sudo*" o argumento "*./*" del archivo *script*, ver en la Figura 3.85.

- *# ./start-arpd.sh*

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd/honeyd_kit-1.0c-a# ./start-ar
pd.sh
+ ./arpd 192.168.1.0/24
arpd[22771]: listening on eth0: arp and (dst net 192.168.1.0/24) and not ether s
rc 08:00:27:ac:20:50
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd/honeyd_kit-1.0c-a#
```

Figura 3.85 Inicialización del *script* de servicio ARP

Red *Honeypot*

La red principal en el desarrollo del módulo se denomina como la Red *Honeypot*. La red principal es una red interna en la dirección "192.168.1.0/24", otorgada y configurada por el demonio de arpd.

La distribución de la red se realiza por medio de la asignación de direcciones, las cuales se encuentran disponibles para 254 *hosts*, debido a la disposición de la máscara 24. Por tal motivo, para el desarrollo del módulo se toma en consideración el manejo de las direcciones IP pertenecientes al rango, ver en la Tabla 3.7.

Tabla 3.7 Disposición de las direcciones IP

Red	Dirección IP	Mascara
Red Principal	192.168.1.0	255.255.255.0
Gateway de Salida	192.168.1.1	255.255.255.0
Ubuntu	192.168.1.15	255.255.255.0
Kali Linux	192.168.1.30	255.255.255.0
Pruebas de Prácticas	192.168.1.18 - 192.168.1.25	255.255.255.0

Inicialización *Honeyd*

Se debe inicializar el demonio de *Honeyd*, el proceso establece el modelo de ejecución del comando principal y los argumentos; archivos de funcionamiento y registro de información. Además, proporcionar la información de equipos que no pueden ser simulados y los diferentes tipos de red a emular [28].

La acción se realiza mediante el comando "*sudo*" o argumento "*./*" del archivo *script*, ver en la Figura 3.86.

- *# ./start-Honeyd.sh*

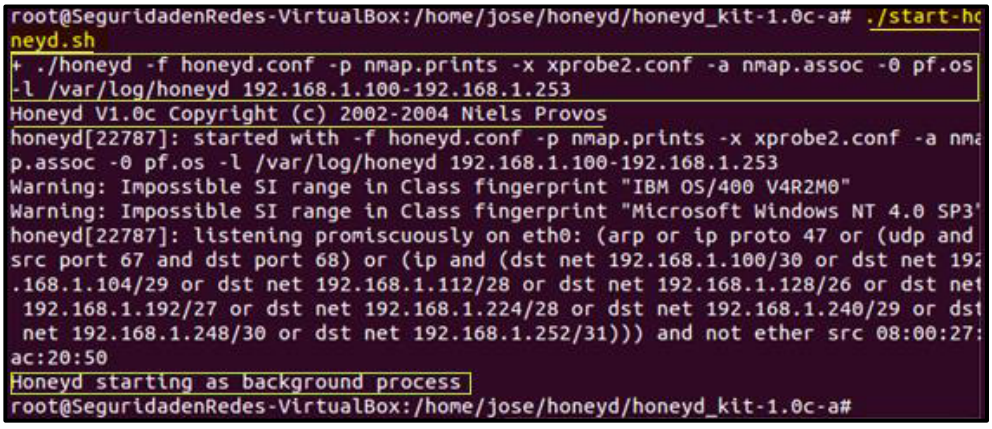


Figura 3.86 Inicialización de *script* de servicio *Honeyd*

Personalidades de *Honeyd*

Los sistemas operativos, elementos de red y *host* son las personalidades que constituyen a *Honeyd*. Las personalidades permiten asignar una identificación en base a un nombre de equipo conformado en cada *Honeypot* sobre la red. Las personalidades se encuentran en los archivos *Nmap.prints* y *Nmap.assoc* [28].

Nmap.prints

Es un archivo de base de datos de huellas digitales de *Nmap OS* constituido en software de *Honeyd* [28].

El ingreso al archivo se lo realiza mediante el comando "*vim*" con el nombre del archivo "*Nmap.prints*", ver en la Figura 3.87.

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd/honeyd_kit-1.0c-a# vim nmap.prints
```

Figura 3.87 Ingreso de archivo *Nmap.prints*

Estructura

El documento se conforma de un listado de personalidades y características de los equipos. El manejo se desarrolla en base a puertos TCP y UDP en función de los paquetes de solicitud o envío [28].

La representación gráfica de una personalidad del archivo *Nmap.prints* se visualiza en la Figura 3.88.

```
# Microsoft Windows NT4 Workstation SP6a
# windows nt4 swith service pack 6
Fingerprint Microsoft Windows NT 4.0 Workstation SP6a
Class Microsoft | Windows | NT/2K/XP | general purpose
TSeq(Class=RI%gcd=<A%SI=<1A66C&>112%IPID=RPI|BI%TS=U)
T1(DF=Y%W=4470|7210|AC00%ACK=S++%Flags=AS%Ops=M)
T2(Resp=Y%DF=N%W=0%ACK=S%Flags=AR%Ops=)
T3(Resp=Y%DF=Y%W=4470|7210|AC00%ACK=S++%Flags=AS%Ops=M)
T4(DF=N%W=0%ACK=O%Flags=R%Ops=)
T5(DF=N%W=0%ACK=S++%Flags=AR%Ops=)
T6(DF=N%W=0%ACK=O%Flags=R%Ops=)
T7(DF=N%W=0%ACK=S++%Flags=AR%Ops=)
PU(DF=N%TOS=0%IPLEN=38%RIPTL=148%RID=E%RIPCK=E%UCK=E%ULEN=134%DAT=E)
```

Figura 3.88 Ejemplo de personalidad en archivo *Nmap.prints* [28]

Los elementos y paquetes de las personalidades del archivo *Nmap.prints* se describen en la Tabla 3.8.

Tabla 3.8 Descripción de paquetes de personalidades en *Nmap.prints* [27] [32]

Elemento	Descripción
Etiqueta	Nombre de equipo o personalidad
TSeq	Prueba de secuencia del protocolo TCP
T1	Paquete SYN con opciones de apertura de puerto
T2	Paquete NULL con opciones de apertura de puerto
T3	Paquete SYN/FIN/URG/PSH con opciones de apertura de puerto
T4	Paquete ACK con opciones de apertura de puerto.
T5	Paquete SYN con opciones de cierre de puerto.
T6	Paquete ACK con opciones de cierre de puerto.
T7	Paquete FIN/PSH/URG con opciones de cierre de puerto.
PU	Paquete UDP para puerto cerrado,

Funciones de paquetes

Descripción de *flags* de segmento TCP en personalidades de *Honeyd***:**

- **SYN.-** Se utiliza para sincronizar números de secuencia durante el establecimiento de la conexión [28].
- **ACK.-** Indica si el número de reconocimiento es significativo. Si el bit está establecido, el número de reconocimiento es el número de secuencia que el remitente espera a continuación, es usado con SYN y FIN [28].
- **RST**.- Reinicia la conexión, que se utiliza principalmente en estados de error [28].
- **FIN**.- Se utiliza para romper una conexión, por lo que no se envían más datos [28].
- **PSH.-** Se encarga de la función de empuje [28].
- **URG**.- Indica si el campo del puntero urgente es significativo [28].

Nmap. Assoc

Es un archivo de base de datos, con asociación de huellas digitales específicas de los sistemas operativos generales de *Honeyd* [28].

El ingreso al archivo se realiza mediante el comando "*vim*" y nombre del archivo "*Nmap.assoc*", ver en la Figura 3.89.

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd/honeyd_kit-1.0c-a# vim nmap.assoc
```

Figura 3.89 Ingreso de archivo *Nmap.assoc* [28]

Listado

En el archivo se dispone de una gran cantidad de nombres de personalidades. Se puede distinguir a las personalidades más importantes, como lo son *Apple MAC OS, GNU/Linux* y *Windows*.

Apple MAC OS

El listado de personalidades *Apple* se encuentra enfocado al manejo de nombres de sistemas operativos, *host* y servidores, ver en la Figura 3.90.

```
Apple Mac OS 8.5.1 (Appleshare IP 6.0);Mac OS X 10.1.5
Apple Mac OS 8.1 running on a PowerPC G3 (iMac);Mac OS X 10.1.5
Apple Mac OS 8.6;Mac OS X 10.1.5
Apple Mac OS 8.6;Mac OS X 10.1.5
Apple Mac OS 9 - 9.1;Mac OS 9.2.x
Apple Mac OS X Server 1.0-1.0-1 (Rhapsody 5.3 - 5.4);Mac OS X 10.1.5
Apple Mac OS X 1.1-1.2 (Rhapsody 5.5-5.6) on a G3;Mac OS X 10.1.5
Apple Mac OS X 10.1 - 10.1.4;Mac OS X 10.1.5
```

Figura 3.90 Listado de personalidades *Apple MAC OS* [28]

GNU/Linux

El listado de personalidades *Linux* se encuentra enfocado al manejo de nombres de sistemas operativos, núcleos y servidores, ver en la Figura 3.91.

```
Linux 2.0.34-38;Linux Kernel 2.4.5 and above
Linux 2.0.39;Linux Kernel 2.4.5 and above
Linux 2.0.35 (S.u.S.E. Linux 5.3 (i386);Linux Kernel 2.4.5 and above
Linux 2.1.24 PowerPC;Linux Kernel 2.4.5 and above
Linux 2.1.76;Linux Kernel 2.4.5 and above
Linux Kernel 2.1.88;Linux Kernel 2.4.5 and above
Linux 2.1.91 - 2.1.103;Linux Kernel 2.4.5 and above
Linux 2.1.19 - 2.2.25;Linux Kernel 2.4.5 and above
```

Figura 3.91 Listado de personalidades *GNU/Linux* [28]

Windows

El listado de personalidades Windows se encuentra enfocado al manejo de nombres de sistemas operativos y servidores, ver en la Figura 3.92.

```
Microsoft Windows XP SP1;Microsoft Windows XP Professional
Microsoft Windows XP Professional SP1;Microsoft Windows XP Professional
Microsoft Windows XP SP1;Microsoft Windows XP Professional
Microsoft Windows XP SP1 or Windows 2000 SP3;Microsoft Windows 2000/2000SP1/2000SP2/2
000SP3
Microsoft Windows XP Professional;Microsoft Windows XP Professional
Microsoft Windows XP SP1;Microsoft Windows XP Professional
Microsoft Windows XP Professional SP1;Microsoft Windows XP Professional
Microsoft Windows XP Professional SP1;Microsoft Windows XP Professional
```

Figura 3.92 Listado de personalidades *Windows* [28]

3.3 Análisis del archivo de configuración

Análisis de Arquitectura de *Honeyd*

Es un diagrama de flujo que permite la visualización general del funcionamiento de *Honeyd*. El proceso radica en el tráfico de la red (*Network*), donde los paquetes entrantes son enrutados (*Routing*) sobre el Despachador de Paquetes Central (*Packet Dispatcher*), el cual se encarga de distribuir correctamente sobre cada protocolo (ICMP, TCP y UDP). En los protocolos TCP y UDP, los servicios (*Services*) configurados se encargan de recibir los datos nuevos y enviar la respuesta en forma de proceso (*Processes*). El tráfico de los paquetes salientes se enruta (*Routing*) hacia el Motor de Personalidad (*Personality Engine*), donde son modificados para imitar el comportamiento de la pila de red configurada de los *Honeypots* sobre la red (*Network*) [28].

La Base de Datos de Configuración (*Configuration Database*) se encarga de almacenar las personalidades configuradas. Además, permite el complemento de interacción realista sobre el Despachador de Paquetes Central, servicios y el Motor de Personalidad [28].

El diagrama de flujo del proceso de *Honeyd* se visualiza en la Figura 3.93.

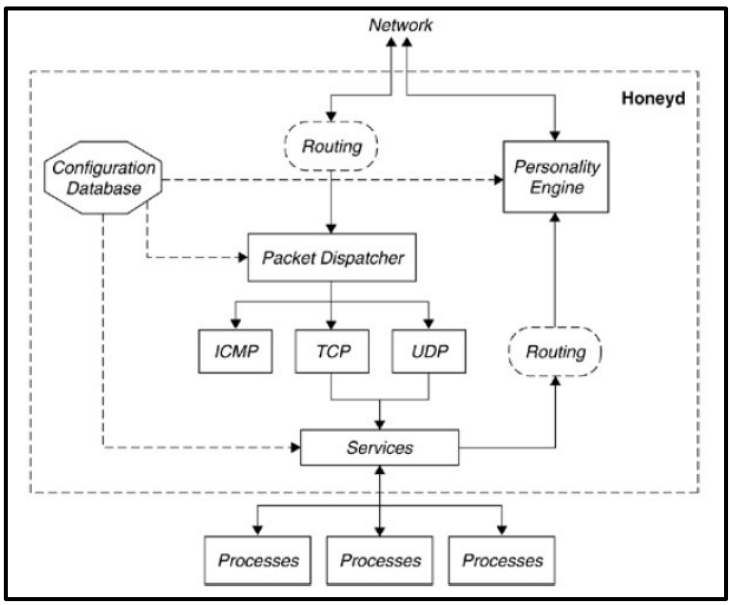


Figura 3.93 Diagrama de flujo del proceso de *Honeyd* [28]

Diseño y adaptabilidad de *Honeyd*

Los sistemas operativos *GNU/Linux* se caracterizan por la configuración de servicios en base a la edición de archivos.

Los archivos que se encargan de la configuración son del tipo texto o denominados "*scripts*", su función radica en el uso de líneas de comandos y ejecución de órdenes para la realización de tareas o acciones sobre un ordenador [40].

La ubicación del archivo de configuración perteneciente *Honeyd* se encuentra en el directorio de la siguiente ruta:

/home/usuario/honeyd/honeyd_kit-1.0c-a/honeyd.conf

El archivo de configuración se caracteriza por la capacidad de personalizar o cambiar de acuerdo con las necesidades específicas. Se encarga de manejar un mecanismo moldeable a las diferentes temáticas o casos establecidos. En base a este archivo, *Honeyd* permite simular una cantidad idónea de 65000 sistemas operativos virtuales sobre una red, estableciendo un señuelo sobre los atacantes [28].

Operaciones del archivo de configuración

Son los mecanismos que constituyen al archivo de configuración para el establecimiento de los sistemas *Honeypots* a la creación de simulación de *host*, sistemas operativos, personalidades y características de red.

Las operaciones se describen a continuación:

- ***create default.-*** Línea de comando para el establecimiento de la identificación de la personalidad [28].
- ***set default personality "Microsoft Windows XP Home Edition".-*** Línea de comando para la asignación de personalidad del sistema operativo virtual. En el archivo *Nmap.assoc* se encuentra la base de datos de las personalidades [28].
- ***set default* TCP *action.-*** Línea de comando de administración de red para establecimiento de funciones de apertura, bloqueo y *reseteo* de los protocolos. El objetivo es proveer un mayor nivel de complejidad a los atacantes. Se establece las palabras "*open"*, "*block"* y "*reset"* respectivamente [28].
- ***add default* TCP *port.-*** Línea de comando de asignación de protocolo y adjudicación de puerto para la interacción de atacante. Se establece en los protocolos TCP y UDP [28].
- ***"sh scripts/misc/test.sh".-*** Ruta de almacenamiento de simulación de servicios en *scripts*. Ejemplos de simulación "*web.sh", "telnetd.sh", "*FTP*.sh"*, etc. [28].

- ***bind 192.168.110.200 default.-*** Línea de comando de asignación de dirección IP a utilizar con el sistema operativo virtual creado [28].

Acciones de protocolos

Son los comportamientos que componen los puertos de los protocolos. A continuación, se describe la acción de cada uno:

Puerto TCP

- ***Open.-*** Respuesta con mensaje *SYN/ACK* de conexión [28].
- ***Block.-*** Pérdida de paquetes sin respuesta, sin conexión [28].
- ***Reset.-*** Respuesta con un *flag RST* de reinicio de conexión [28].
- ***Tarpit.-*** Respuesta de añadimiento de conexión [28].

Puerto UDP

- ***Open.-*** Sin Respuesta [28].
- ***Block.-*** Pérdida de paquetes sin respuesta [28].
- **Reset.-** Respuesta con un puerto y mensaje ICMP de error [28].

Puerto ICMP

- ***Open.-*** Respuesta con paquetes de mensaje ICMP [28].
- ***Block.-*** Pérdida de paquetes sin respuesta [28].

Contenido del archivo de configuración "*honeyd.conf*"

El archivo de configuración se compone de cada uno de los elementos previamente revisados. El objetivo principal del archivo es proporcionar los mecanismos para la elaboración de los equipos virtuales de *Honeypots* [28].

La representación gráfica del índice del archivo de configuración y los detalles referentes a la fecha de actualización y acciones por defecto se visualiza en la Figura 3.94.

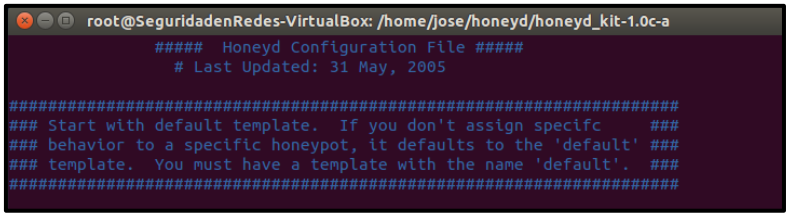


Figura 3.94 Índice de archivo de configuración de *Honeyd* [28]

La asignación de las operaciones del archivo de configuración se describe en la plantilla por defecto. Se describen las líneas de comandos, las cuales se encargan del manejo

de los nombres de personalidades, acciones de protocolos, puertos, servicios y direcciones IP [28].

La representación gráfica de la plantilla por defecto de *Honeyd* se visualiza en la Figura 3.95.

```
### Default Template
create default
set default personality "Microsoft Windows XP Home Edition"
set default default tcp action reset
set default default udp action reset
set default default icmp action open
add default tcp port 80 "sh scripts/misc/test.sh"
add default tcp port 139 open
add default tcp port 137 open
add default udp port 137 open
add default udp port 135 open
bind 192.168.110.200 default
```

Figura 3.95 Plantilla de archivo de configuración de *Honeyd* [28]

Las acciones de protocolos se describen en el archivo de configuración, con el fin de establecer el comportamiento de cada puerto. Se desglosan el procedimiento de los puertos TCP, UDP e ICMP, para el manejo de las distintas personalidades empleadas con *Honeyd* en base a los sistemas *Honeypots* [28]..

La representación gráfica de las acciones y comportamiento de los protocolos se visualiza en la Figura 3.96.

```
#####################################################################
### We now have the rest of our templates and honeypot behavior   ###
###                                                               ###
### Port Behavior                                                 ###
### TCP (default is Open)                                         ###
###  - Open:   Respond with Syn/Ack, establish connection         ###
###  - Block:  Drop packet and do not reply                       ###
###  - Reset:  Respond with RST                                   ###
###  - Tarpit: Sticky connection                                  ###
###                                                               ###
### UDP (default is Closed)                                       ###
###  - Open:  No response                                         ###
###  - Block: Drop packet and do not reply                        ###
###  - Reset: Respond with ICMP port error message                ###
###                                                               ###
### ICMP (default is Open)                                        ###
###  - Open:  Reply to ICMP packets                               ###
###  - Block: Drop packet and do not reply                        ###
###                                                               ###
### Ethernet Codes                                                ###
###  - Located in ethernet.c file, compiled into honeyd           ###
#####################################################################
```

Figura 3.96 Acciones de protocolo del archivo de configuración [28]

3.4 Hojas guías de prácticas

En base al proyecto se detallan las hojas guías para docentes y estudiantes para el desarrollo de las prácticas:

PRÁCTICA 1

1. **TEMA:** Introducción a los *Honeypots*
2. **DESARROLLO DE LA PRÁCTICA**

NOTA: El tiempo para realizar la práctica es de 25 minutos. Se requiere que el instructor explique el diseño, adaptadores y configuración de la Red *Honeypot*.

DISEÑO DE RED

Un *Honeypot* es un sistema señuelo encargado de la seguridad informática dentro de una red. Se convierte en el objetivo de un posible ataque informático para que pueda ser detectado y se pueda obtener información de él y del atacante.

El desarrollo del módulo se llevará a cabo en el programa *Oracle VirtualBox*. El diseño de red se realiza en base a la Red *Honeypot* en la dirección 192.168.1.0/24. El equipo principal utilizado es *Ubuntu*, el cual mediante la herramienta *Honeyd* se procede a emular un *Honeypot* simple. Asimismo, cuenta con un equipo verificador o atacante denominado *Kali Linux*. La topología de red se visualiza en la Figura 3.97.

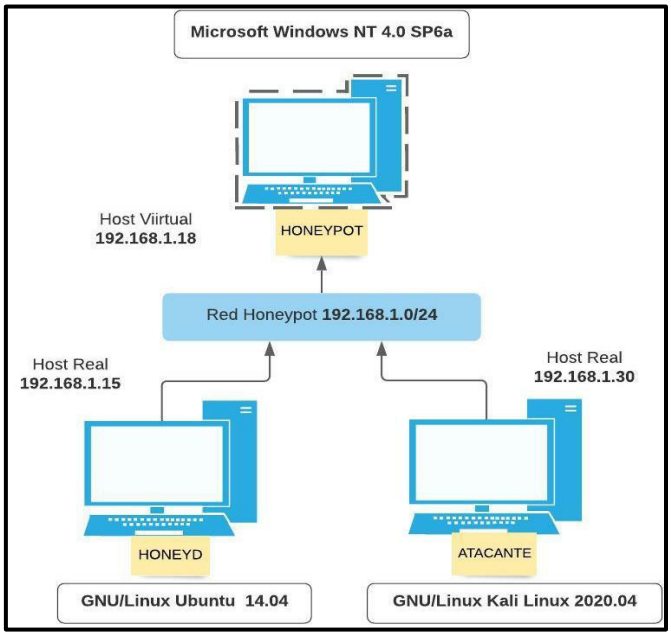


Figura 3.97 Topología de Práctica No.1

Paso 1: Validación de Red

Se debe validar que la máquina se encuentre en red mediante la opción "Configuración" en la máquina virtual *Ubuntu,* ver en la Figura 3.98..

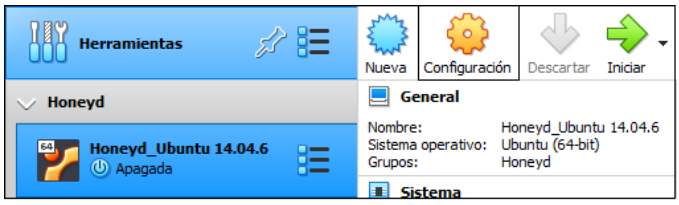


Figura 3.98 Ingreso a la opción "Configuración"

Se opta por validar que se encuentre activado el adaptador de red 1 en red interna denominado "red1", ver en la Figura 3.99.

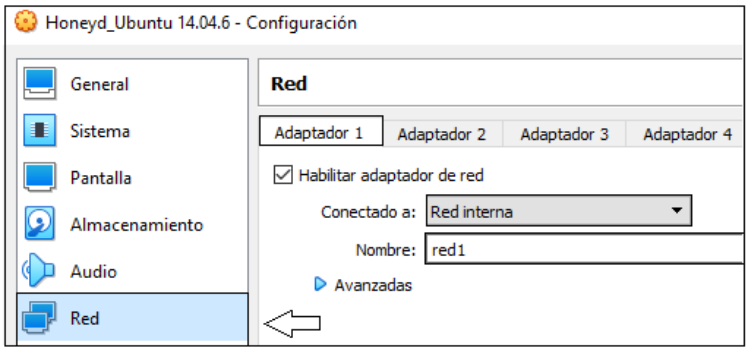


Figura 3.99 Validación de adaptador 1 en "red1"

Asimismo, se encuentre habilitado el adaptador de red 2 en red NAT, ver en la Figura 3.100.

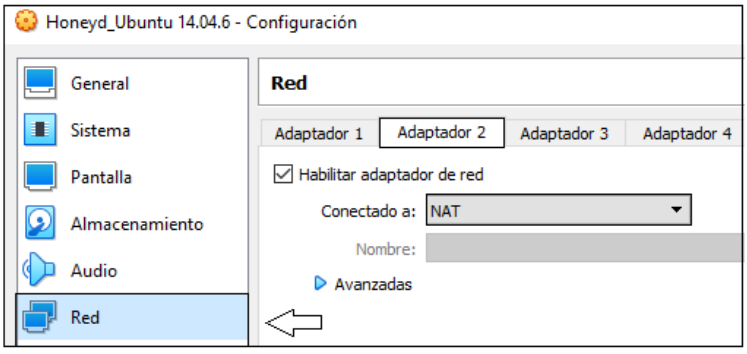


Figura 3.100 Validación de adaptador 2 en "NAT"

Paso 2: Se procede a ingresar en la máquina virtual *Ubuntu*, ver en la Figura 3.101.

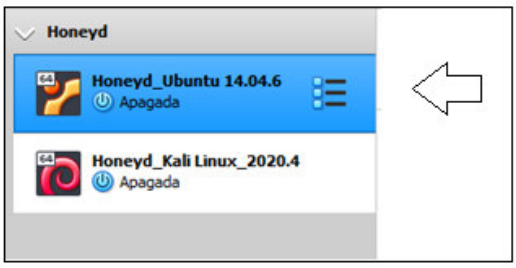


Figura 3.101 Máquina virtual de *Honeyd* en *Ubuntu*

Se ingresa al sistema operativo mediante la clave "*jose*" y se procede a abrir en el ícono de la terminal, ver en la Figura 3.102.

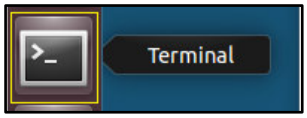


Figura 3.102 Ícono de terminal

Paso 3: Se debe ingresar a la línea de comandos y ejecutar el directorio principal de *Honeyd* con la descripción de los pasos, ver en la Figura 3.103.

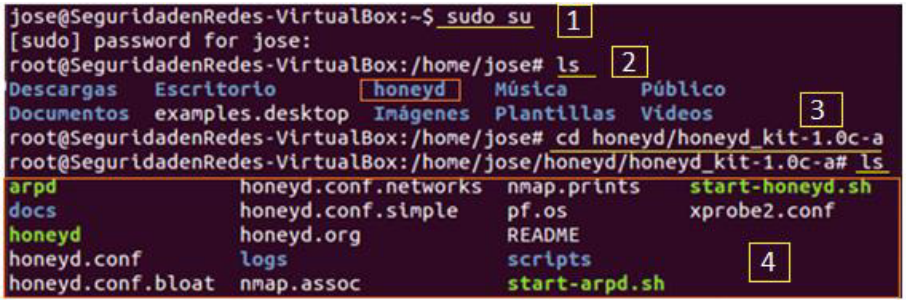


Figura 3.103 Ingreso al directorio principal de *Honeyd* 1

1. Ingreso al usuario root con privilegios mediante el comando "*sudo su*" e ingreso con la clave" *jose*".
2. Visualización del directorio "*Honeyd*" mediante el comando "*ls*".
3. Ingreso al directorio principal de *Honeyd* mediante el comando "*cd*" y la ruta "*honeyd/honeyd_kit-1.0c-a*".
4. Visualización de los componentes de *Honeyd* mediante el comando "*ls*".

Paso 4: Elaboración del archivo de configuración con la descripción de los pasos, ver en la Figura 3.104.

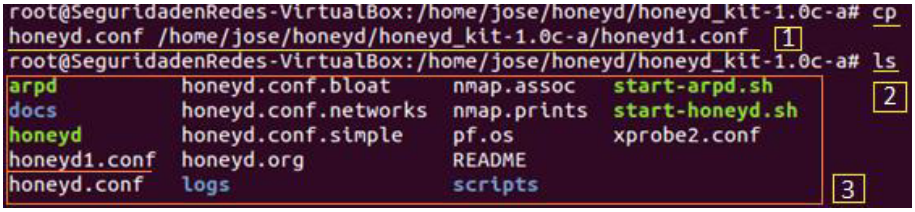


Figura 3.104 Elaboración de archivo de configuración "*honeyd1.conf*"

1. Se realiza una copia del archivo "*honeyd.conf*" con el nombre "*honeyd1.conf*" mediante el comando "*cp"* y la ruta del directorio principal de *Honeyd.*
2. Visualización de archivo creado mediante el comando "*ls*".
3. El archivo "*honeyd1.conf*" en listado de archivos de *Honeyd*.

Paso 5: Verificación de la tarjeta de red "eth0" y dirección IP en red interna mediante el comando "*ifconfig*", ver en la Figura 3.105.

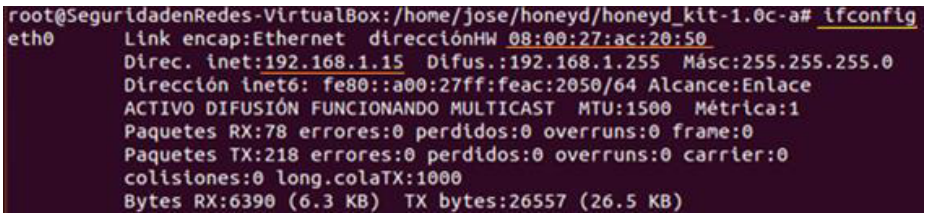


Figura 3.105 Validación de tarjeta de red y dirección IP

Paso 6: Selección del sistema operativo.

Se debe seleccionar la personalidad *Honeypot* a emular en la Red *Honeypot* con la descripción de los pasos, ver en la Figura 3.106.

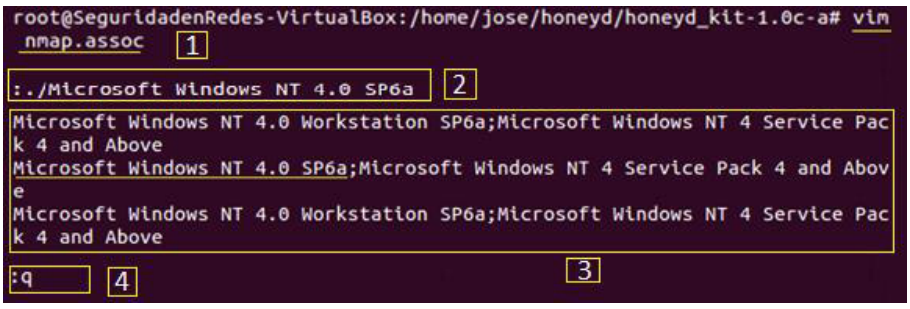


Figura 3.106 Selección de personalidad en Práctica No.1

1. Ingreso al archivo de base de datos de personalidades mediante el comando *"vim*" y nombre del archivo "*Nmap.assoc*".
2. Búsqueda de personalidad mediante "Esc" y el argumento *":./Nombre de Personalidad*".

3. Listado de personalidades y selección de la personalidad indicada en el diseño de red.
4. Salida de archivo mediante “Esc + q”.

Paso 7: Análisis de archivo de configuración “*Honeyd1.conf*”.

Ingreso al archivo de configuración “*honeyd1.conf*” mediante el comando “*vim*”, ver en la Figura 3.107.

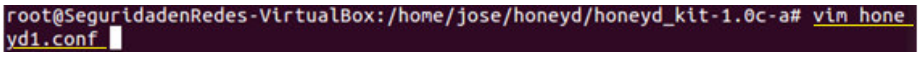


Figura 3.107 Ingreso a archivo de configuración “*honeyd1.conf*”

Análisis de plantilla por defecto del archivo de configuración, ver en la Figura 3.108.

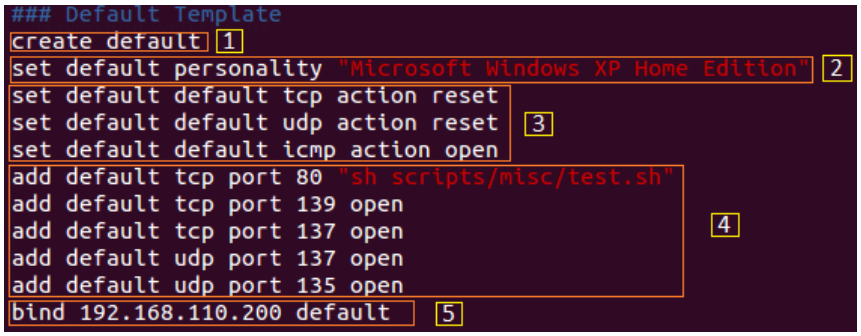


Figura 3.108 Características de plantilla por defecto en archivo *“honeyd.conf*”

1. Línea de comando de creación de personalidad por medio de la opción “*create*”.
2. Línea de comando de establecimiento de etiqueta por medio de la opción “*set*”.
3. Línea de comando de establecimiento de acciones en protocolos TCP, UDP e ICMP. Adjudicando acciones de *reseteo*, bloqueo y apertura mediante las opciones “*reset*, *block* y *open*” respectivamente.
4. Línea de comando de establecimiento de puertos de protocolos TCP y UDP. Establecimiento de ruta de archivo del tipo *script* para simulación de servicios por medio de la opción “*add*”.
5. Línea de comando de establecimiento de dirección IP de personalidad en base a la opción “*bind*”.

Paso 8: Configuración del archivo “*honeyd1.conf*”.

Nota: El establecimiento de la configuración se realiza en base al diseño de red y la plantilla por defecto en el apartado final del archivo. La descripción de los pasos se visualiza en la Figura 3.109.

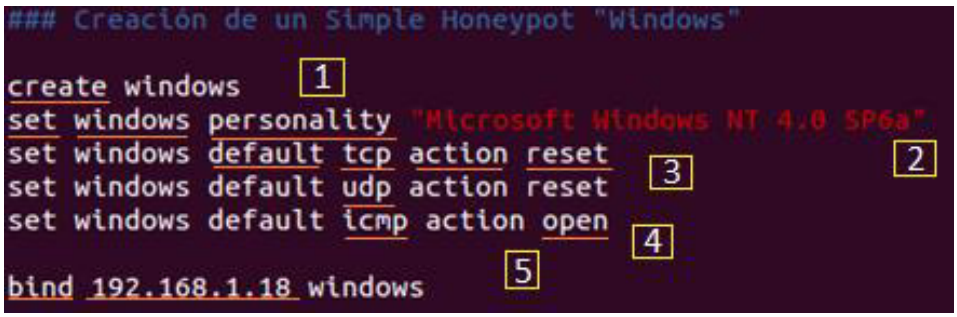


Figura 3.109 Configuración del archivo "*honeyd1.conf*"

1. Asignación de personalidad "windows" por opción "*create*".
2. Asignación de personalidad "*Microsoft Windows NT 4.0 SP6a*" por opción "*set*".
3. Asignación de acción de protocolos TCP y UDP por la opción predetermina "*reset*".
4. Habilitación de mensajes ICMP por opción "*open*".
5. Establecimiento de dirección IP de *Honeypot* por opción "*bind*" en "*windows*".

Paso 9: Levantamiento de herramientas de *Honeyd.*

Se inicia el proceso de levantamiento de servicio "*arpd*" mediante la opción de comando *"./start-arpd.sh*" para la inicialización del proceso ARP para la salida de red del *Honeypot*, ver en la Figura 3.110.

Figura 3.110 Inicialización del proceso ARP en Práctica No.1

Se inicia el proceso *Honeyd* mediante la opción de comando "*./start-honeyd.sh*" para la inicialización del servicio, ver en la Figura 3.111.

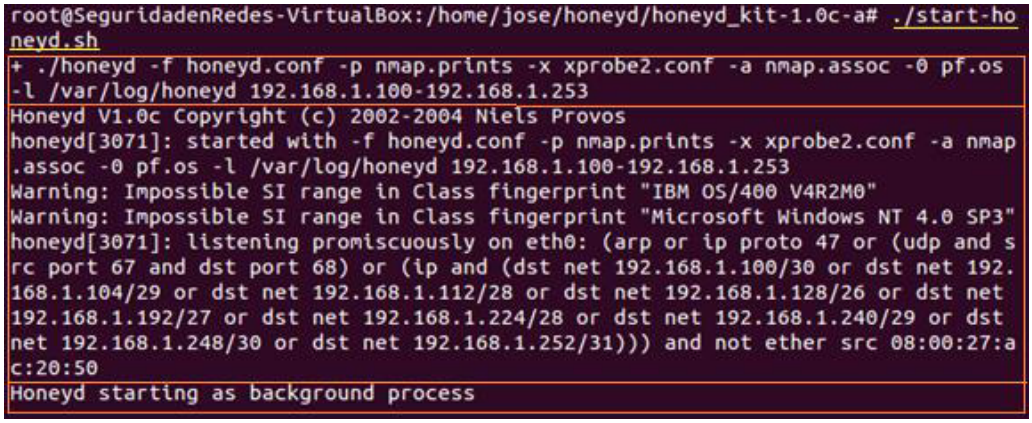


Figura 3.111 Inicialización del proceso *Honeyd* en Práctica No.1

Paso 10: Inicio de ejecución del servicio de *Honeyd.*

La inicialización se realiza mediante el comando "*honeyd*" acompañado de argumentos y opciones, ver en la Tabla 3.9.

Tabla 3.9 Comandos de inicialización de *Honeyd* en Práctica No.1 [27] [32]

Opción	Función
honeyd	Arranque de *Honeyd*
-d	Levantamiento del Demonio
-i	Tarjeta de Red "eth0"
-f	Archivo "*Honeyd.conf1*"- ruta
-p	Archivo "*Nmap.prints*" conjunto de personalidades-ruta
-x	Archivo "*xprobe.conf*" sonda-lector de personalidades-ruta
-a	Archivo "*Nmap.assoc*" personalidades-sonda-ruta
-0	Archivo "*pf.os*" lector de personalidades pasivas-ruta
-l	Archivo de Registro de Información
192.168.1.18	Dirección IP del *Honeypot*

Se realiza la implementación del comando con los argumentos, el demonio de *Honeyd* muestra los argumentos, advertencias y procesos del servicio, ver en la Figura 3.112.

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd/honeyd_kit-1.0c-a# honeyd -d
-i eth0 -f /home/jose/honeyd/honeyd_kit-1.0c-a/honeyd1.conf -p /home/jose/honeyd
/honeyd_kit-1.0c-a/nmap.prints -x /home/jose/honeyd/honeyd_kit-1.0c-a/xprobe2.co
nf -a /home/jose/honeyd/honeyd_kit-1.0c-a/nmap.assoc -0 /home/jose/honeyd/honeyd
_kit-1.0c-a/pf.os -l /var/log/honeyd/honeyd1.log 192.168.1.18
Honeyd V1.5c Copyright (c) 2002-2007 Niels Provos
honeyd[2708]: started with -d -i eth0 -f /home/jose/honeyd/honeyd_kit-1.0c-a/hon
eyd1.conf -p /home/jose/honeyd/honeyd_kit-1.0c-a/nmap.prints -x /home/jose/honey
d/honeyd_kit-1.0c-a/xprobe2.conf -a /home/jose/honeyd/honeyd_kit-1.0c-a/nmap.ass
oc -0 /home/jose/honeyd/honeyd_kit-1.0c-a/pf.os -l /var/log/honeyd/honeyd1.log 1
92.168.1.18
Warning: Impossible SI range in Class fingerprint "IBM OS/400 V4R2M0"
Warning: Impossible SI range in Class fingerprint "Microsoft Windows NT 4.0 SP3"
honeyd[2708]: listening promiscuously on eth0: (arp or ip proto 47 or (udp and s
rc port 67 and dst port 68) or (ip and (host 192.168.1.18))) and not ether src 0
8:00:27:ac:20:50
honeyd[2708]: Demoting process privileges to uid 65534, gid 65534
```

Figura 3.112 Ejecución de *Honeyd* en Práctica No.1

Paso 11: Inicialización de *Kali Linux*

Se procede a inicializar la máquina *Kali Linux* antes de iniciar el demonio de *Honeyd*. Se debe validar el *Honeypot* creado mediante la máquina *Kali Linux*. Se comprueba la tarjeta de red en la opción “Configuración”, ver en la Figura 3.113.

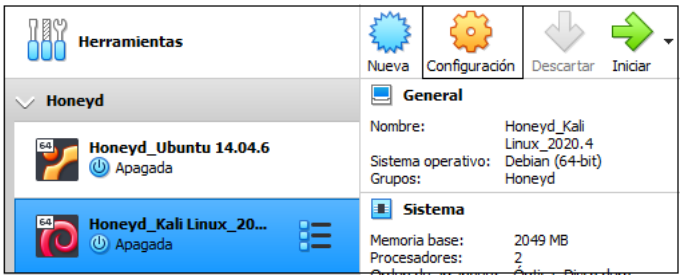


Figura 3.113 Ingreso a opción “Configuración” en *Kali Linux*

Se procede a validar que se encuentre activado el adaptador de red 1 en red interna denominado “red1”, ver en la Figura 3.114.

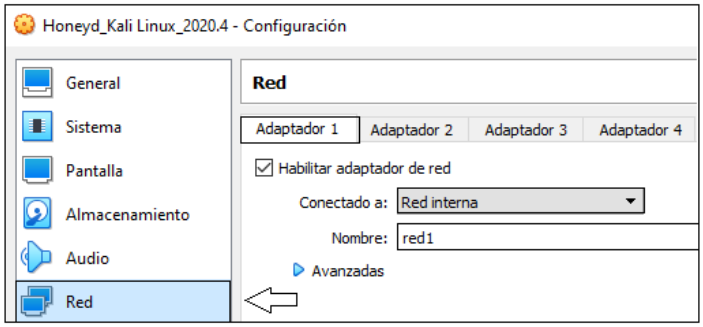


Figura 3.114 Validación de adaptador 1 en “red1”

Asimismo, se encuentre habilitado el adaptador de red 2 en red NAT, ver en la Figura 3.115.

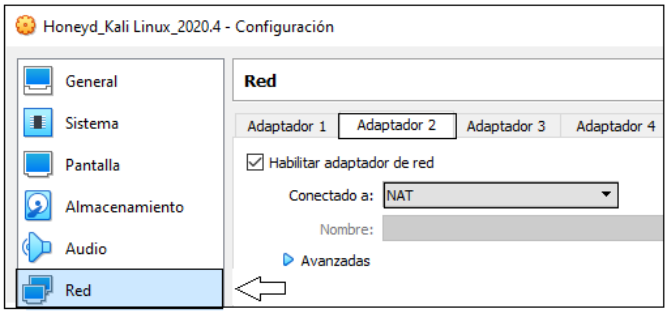


Figura 3.115 Validación de adaptador 2 en *"NAT"*

Se procede a ingresar en la máquina virtual *Kali Linux*, ver en la Figura 3.116.

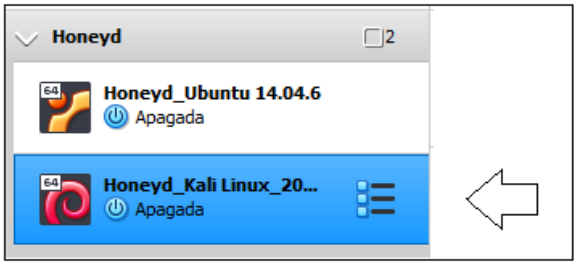


Figura 3.116 Ingreso a equipo *Kali Linux*

Se procede a ingresar al sistema operativo mediante las credenciales.

Nota: El ingreso de las credenciales se debe realizar en el menor tiempo posible para que el equipo se prepare para la ejecución de la red interna

- **Usuario**: *jose*
- **Clave**: *josemegadeth*

Se validan las conexiones de las tarjetas de red 1 y 2, ver en la Figura 3.117.

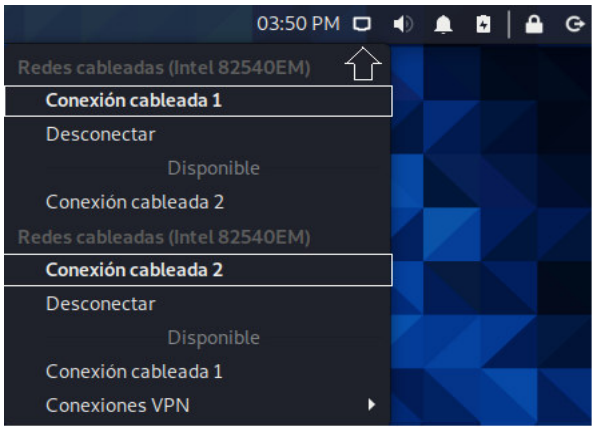


Figura 3.117 Disposición de tarjetas de red sobre el equipo *Kali Linux*

Se opta por abrir la terminal en el icono ubicado en el lado superior izquierdo, ver en la Figura 3.118.

Figura 3.118 Icono de visualización de terminal en *Kali Linux*

Se debe ingresar como usuario "*root*" mediante el comando "*sudo su*" y se ingresa la clave de usuario previamente utilizada. Se valida el estado de la tarjeta de red en base a la dirección IP asignada en el diseño de red, ver en la Figura 3.119.

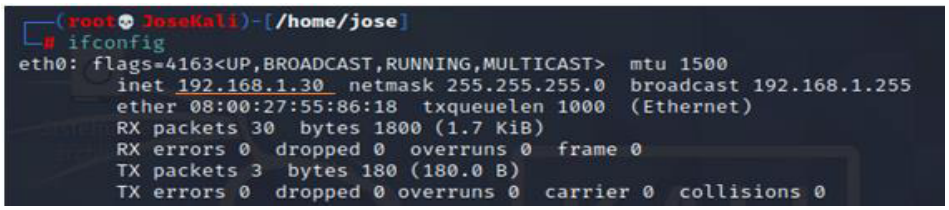


Figura 3.119 Validación de tarjetas de red en la dirección IP en *Kali Linux*

Paso 12: Verificación de *Red Honeypot*

Se procede a verificar el *Honeypot* creado mediante el comando "*ping*" en la máquina *Kali Linux*, ver en la Figura 3.120.

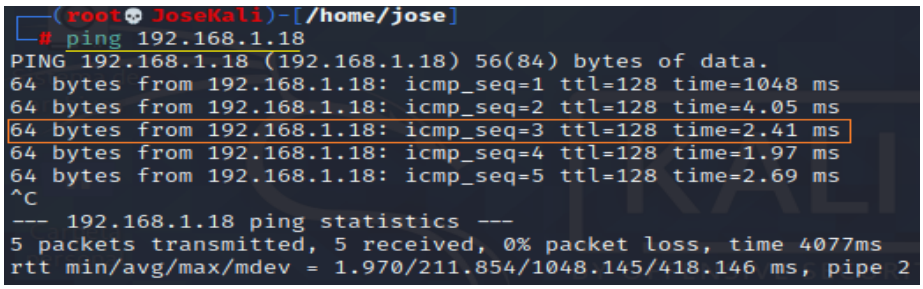


Figura 3.120 Comando "*ping*" en verificación de *Honeypot* en Práctica No.1.

Se debe cerrar el proceso del comando "*ping*", realizando la acción en el teclado "Ctrl+ c".

El establecimiento de la conexión entre máquinas se realiza mediante los mensajes ICMP. Se realiza el proceso de envío de mensajes desde el *Honeypot* 192.168.1.18 hacia el equipo externo 192.168.1.30 y valida el *host* virtual, ver en la Figura 3.121.

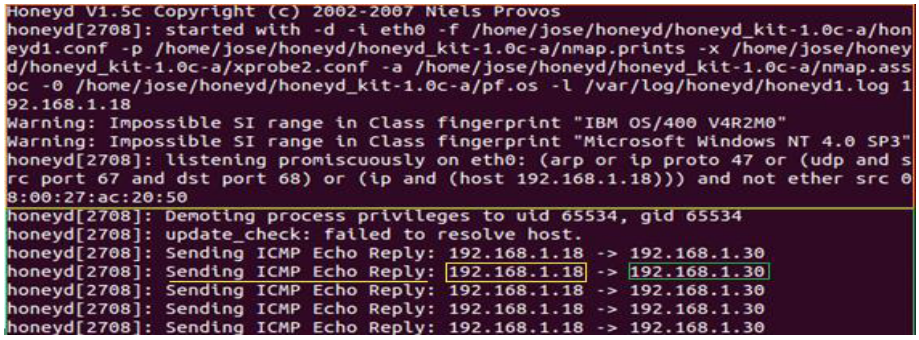


Figura 3.121 Comprobación de conexión de *Honeypot* con *Kali Linux*

Se opta por cerrar el proceso de *Honeyd*, realizando la acción en el teclado "Ctrl+ c".

Paso 13: Archivo de registro de información

Se realiza el ingreso del archivo de registro mediante el comando "*vim*" acompañado de la ruta "*/var/log/honeyd/* "con el nombre del archivo "*honeyd1.log*", ver en la Figura 3.122.

Figura 3.122 Directorio de almacenamiento de información Práctica No.1

El archivo muestra el análisis y los datos proporcionados por *Honeyd* a la respuesta del *Honeypot* "*Microsoft Windows NT 4.0 SP6*", ver en la Figura 3.123.

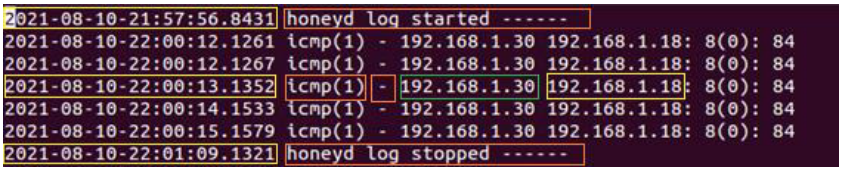


Figura 3.123 Archivo de registro de información de Práctica No.1

1. Fecha y hora de la actividad realiza.
2. Mensaje "ICMP" enviado entre *hosts*.
3. Opción (-) sin conexión de puerto.
4. Dirección IP del origen 192.168.1.30 (*Kali Linux*) con la dirección IP del destino 192.168.1.18 (*Honeypot*).
5. Fecha y hora de la actividad finalizada.

Nota: El primer ejemplo muestra el formato de información proporcionado por *Honeyd* en base a la interacción del *Honeypot* (*Microsoft Windows NT 4.0 SP6*). La elección de la personalidad puede ser asignada arbitrariamente.

3. CONCLUSIONES

- En el desarrollo de las tecnologías de la información y redes, realizar actividades de software en sistemas de virtualización permite ahorrar los recursos físicos del equipo real y a su vez permite distribuir el espacio del almacenamiento eficientemente.
- Los sistemas de seguridad informática son indispensables en el mundo actual debido a las grandes tasas de infiltración dentro de los equipos de la red, razón por la cual tener equipos señuelos disminuye el riesgo de ser atacados por un agente externo.
- El software de código abierto de *Honeyd* se basa en un archivo de configuración básico, capaz de proporcionar a la red un *host* virtual (*Honeypot*) con la capacidad de poder interactuar los elementos disponibles en la red.
- En el mundo de las redes de comunicación, el comando "*ping*" es el mecanismo que establece la comunicación entre los equipos mediante una dirección IP o una dirección de dominio.

4. RECOMENDACIONES

- Se sugiere validar las conexiones de red del equipo *Kali Linux* con el fin de evitar cualquier tipo de problema que se pudiese generar en el desarrollo de la práctica.
- Se recomienda ejecutar el servicio de *Honeyd* una vez que se encuentre habilitadas las funciones del equipo *Kali Linux*, con el fin de evitar errores o posibles congelamientos de pantalla.
- Se recomienda utilizar una dirección IP de acuerdo al rango de direcciones disponibles en la *Red Honeypot*. Debido a que, si se realiza la configuración con otro tipo de red, no existirá ningún tipo de conexión o verificación de funcionamiento.
- Se recomienda leer y realizar cada uno de los pasos detenidamente. Cabe mencionar que la elección de la personalidad es a modo de validación de la existencia de está. Sin embargo, el procedimiento muestra la forma correcta de emplear en el archivo configuración.

PRÁCTICA 2

1. **TEMA:** Validación de Puertos TCP y UDP
2. **DESARROLLO DE LA PRÁCTICA**

NOTA: El tiempo para realizar la práctica es de 25 minutos. Se requiere que el instructor explique los lineamientos de la Red *Honeypot*. Además, es importante que el estudiante realice el trabajo preparatorio de acuerdo con los lineamientos solicitados.

DISEÑO DE RED

El diseño de red se realiza en base a la Red *Honeypot* en la dirección 192.168.1.0/24. El equipo principal utilizado es *Ubuntu*, el cual mediante la herramienta *Honeyd* se procede a emular un *Honeypot*. El *host* virtual se configura mediante la personalidad y puertos TCP y UDP. Además, se cuenta con equipo verificador o atacante denominado *Kali Linux.* La topología de red se visualiza en la Figura 3.124.

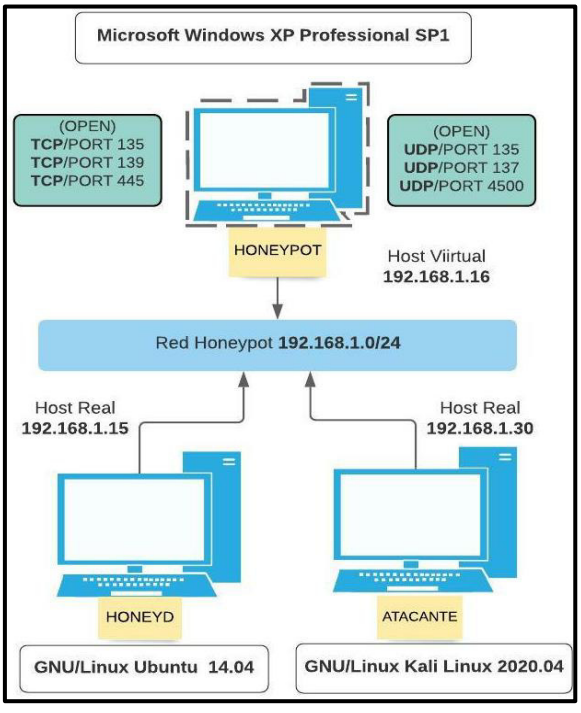


Figura 3.124 Topología de Práctica No.2

Nota: Tomar en cuenta las mismas consideraciones en cuanto a tarjeta de red, ingreso de máquinas virtuales y manejo de terminal detallado en Práctica No.1**.**

Paso 1: Elaboración del archivo de configuración con la descripción de los pasos, ver en la Figura 3.125.

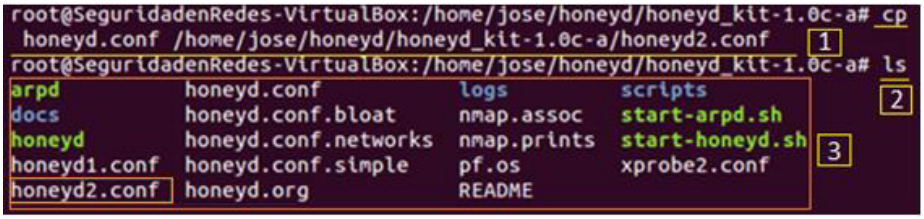


Figura 3.125 Elaboración del archivo configuración "*honeyd2.conf*"

1. Se procede a realizar una copia del archivo "*honeyd.conf*" con el nombre "*honeyd2.conf*" mediante el comando "*cp*" y la ruta del directorio principal de *Honeyd*.
2. Visualización de archivo creado mediante el comando "*ls*".
3. El archivo "*honeyd2.conf*" en listado de archivos de *Honeyd*.

Paso 2: Selección del sistema operativo.

Se procede a seleccionar la personalidad *Honeypot* a emular en la Red *Honeypot* con la descripción de los pasos, ver en la Figura 3.126.

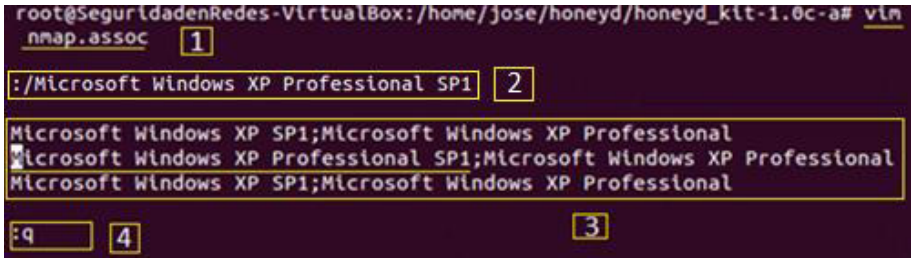


Figura 3.126 Selección de personalidad en Práctica No.2

1. Ingreso al archivo de base de datos de personalidades mediante el comando "*vim*" y nombre del archivo "*Nmap. assoc*".
2. Búsqueda de personalidad mediante "Esc" y el argumento *":/Nombre de Personalidad*".
3. Listado de personalidades y selección de la personalidad indicada en el diseño de red.
4. Salida de archivo mediante "Esc + q".

Paso 3: Ingreso al archivo de configuración "*honeyd2.conf*".

Ingreso al archivo de configuración "*honeyd2.conf*" mediante el comando "*vim*", ver en la Figura 3.127.

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd/honeyd_kit-1.0c-a# vim
 honeyd2.conf
```

Figura 3.127 Ingreso al archivo de configuración "*honeyd2.conf*"

Paso 4: Configuración del archivo "*honeyd2.conf*".

Nota: El establecimiento de la configuración se realiza en base al diseño de red y la plantilla por defecto en el apartado final del archivo. La descripción de los pasos se visualiza en la Figura 3.128.

```
### Creación de plantilla de "Honeypot" y puertos

create windows   1

set windows personality "Microsoft Windows XP Professional SP1"
set windows default tcp action reset
set windows default udp action reset   3                    2
set windows default icmp action open
add windows tcp port 139 open          4
add windows tcp port 445 open   5
add windows tcp port 135 open
add windows udp port 135 open
add windows udp port 137 open   6
add windows udp port 4500 open

bind 192.168.1.16 windows   7
```

Figura 3.128 Configuración del archivo "*honeyd2.conf*"

1. Asignación de personalidad "windows" por la opción "*create*".
2. Asignación de personalidad "*Microsoft Windows XP Professional SP1*" por opción "*set*".
3. Asignación de acción de protocolos TCP y UDP por opción predeterminada "*reset*".
4. Habilitación de mensajes ICMP por opción "*open*".
5. Asignación de puertos del Protocolo TCP por la opción "*add*".
6. Asignación de puertos del Protocolo UDP por la opción "*add*".
7. Establecimiento de dirección IP de *Honeypot* por la opción "*bind*" en "windows".

Paso 5: Ejecución de servicios de inicialización

Se procede a ejecutar los archivos *scripts* de los servicios ARP y *Honeyd*:

- *"./start-arpd.sh*
- "*./start-honeyd.sh*"

Nota: Tomar en consideración el procedimiento de ejecución de los *scripts* ARP y *Honeyd*, realizado en la Práctica No.1.

Paso 6: Inicio de ejecución del servicio de *Honeyd*

La inicialización se realiza mediante el comando "*honeyd*" acompañado de argumentos y opciones, ver en la Tabla 3.10.

Tabla 3.10 Comandos de inicialización de *Honeyd* en Práctica No.2 [27] [32]

Opción	Función
honeyd	Arranque de *Honeyd*
-d	Levantamiento del Demonio
-i	Tarjeta de Red "eth0"
-f	Archivo "*Honeyd.conf2*"- ruta
-p	Archivo "*Nmap.prints*" conjunto de personalidades-ruta
-x	Archivo "*xprobe.conf*" sonda-lector de personalidades-ruta
-a	Archivo "*Nmap.assoc"* personalidades-sonda-ruta
-0	Archivo "*pf.os*" lector de personalidades pasivas-ruta
-l	Archivo de Registro de Información
192.168.1.16	Dirección IP del *Honeypot*

Se realiza la implementación del comando con los argumentos, el demonio de *Honeyd* muestra los argumentos, advertencias y procesos del servicio, ver en la Figura 3.129.

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd/honeyd_kit-1.0c-a# honeyd -d
-i eth0 -f /home/jose/honeyd/honeyd_kit-1.0c-a/honeyd2.conf -p /home/jose/honeyd
/honeyd_kit-1.0c-a/nmap.prints -x /home/jose/honeyd/honeyd_kit-1.0c-a/xprobe2.co
nf -a /home/jose/honeyd/honeyd_kit-1.0c-a/nmap.assoc -0 /home/jose/honeyd/honeyd
_kit-1.0c-a/pf.os -l /var/log/honeyd/honeyd2.log 192.168.1.16
Honeyd V1.5c Copyright (c) 2002-2007 Niels Provos
honeyd[3360]: started with -d -i eth0 -f /home/jose/honeyd/honeyd_kit-1.0c-a/hon
eyd2.conf -p /home/jose/honeyd/honeyd_kit-1.0c-a/nmap.prints -x /home/jose/honey
d/honeyd_kit-1.0c-a/xprobe2.conf -a /home/jose/honeyd/honeyd_kit-1.0c-a/nmap.ass
oc -0 /home/jose/honeyd/honeyd_kit-1.0c-a/pf.os -l /var/log/honeyd/honeyd2.log 1
92.168.1.16
Warning: Impossible SI range in Class fingerprint "IBM OS/400 V4R2M0"
Warning: Impossible SI range in Class fingerprint "Microsoft Windows NT 4.0 SP3"
honeyd[3360]: listening promiscuously on eth0: (arp or ip proto 47 or (udp and s
rc port 67 and dst port 68) or (ip and (host 192.168.1.16))) and not ether src 0
8:00:27:ac:20:50
honeyd[3360]: Demoting process privileges to uid 65534, gid 65534
```

Figura 3.129 Inicialización del demonio de *Honeyd* en Práctica No.2

Paso 7: Validación de *Honeypot* en *Kali Linux.*

Nota: Validar las conexiones de las tarjetas de red y configuración de la dirección IP del equipo *Kali Linux* de acuerdo a la Practica No.1.

Se procede a verificar el *Honeypot* creado mediante el comando "*ping*" en la máquina *Kali Linux*, ver en la Figura 3.130.

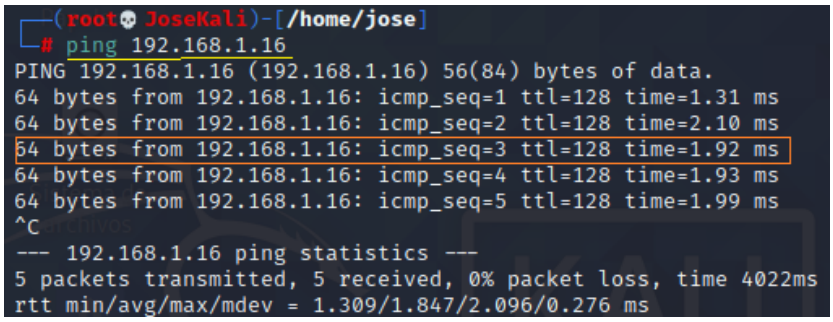


Figura 3.130 Comando "*ping*" en verificación de *Honeypot* en Práctica No.2

Se procede a cerrar el proceso del comando "*ping*" realizando la acción en el teclado "Ctrl+ c".

El establecimiento de la conexión entre máquinas se realiza mediante los mensajes ICMP. Se realiza el proceso de envío de mensajes desde el *Honeypot* 192.168.1.16 hacia el equipo externo 192.168.1.30 y valida el *host* virtual, ver en la Figura 3.131.

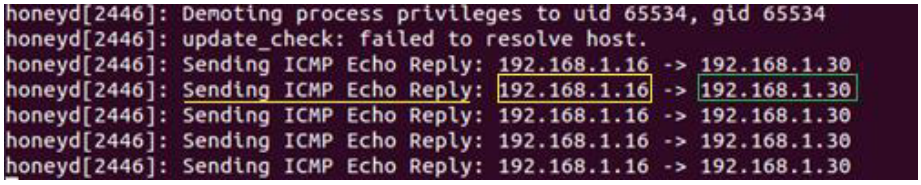


Figura 3.131 Establecimiento de conexión de *Honeypot* en Práctica No.2

En *Kali Linux* se procede a escanear el *host* mediante la herramienta *Nmap*. El comando "*Nmap*" se acompaña de la opción "*--send-ip -P0*" para el análisis de los puertos TCP, ver en la Figura 3.132.

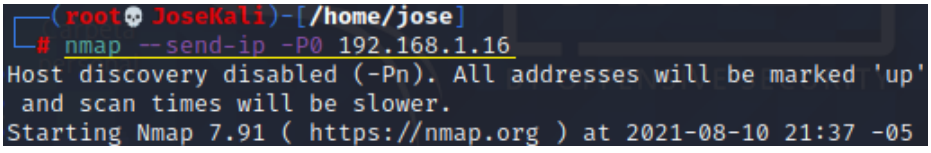


Figura 3.132 Escaneo de puertos TCP con *Nmap* en Práctica No.2

Honeyd notifica el escaneo de la red en los puertos TCP. Analiza la dirección IP del origen con el puerto asignado hacia la dirección IP del destino con puerto correspondiente. El mensaje de interacción de *Honeyd* es "*Killing attempted connection"* al español "Intento de conexión eliminado", ver en la Figura 3.133.

Figura 3.133 Comportamiento de escaneo puertos TCP en Práctica No.2

Cuando finaliza el mapeo, *Nmap* se encarga de ilustrar los resultados de las operaciones realizadas dentro del *Honeypot*. Muestra los puertos TCP otorgados y definidos en el archivo de configuración, ver en la Figura 3.134.

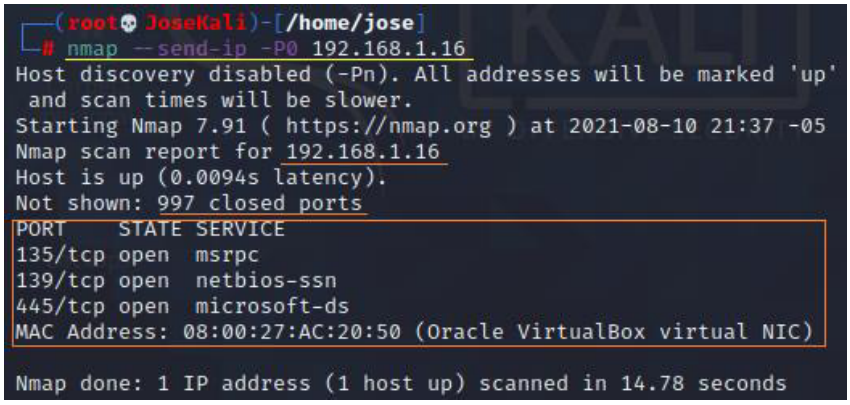


Figura 3.134 Resultados de escaneo puertos TCP en Práctica No.2

Del mismo modo se procede a realizar el escaneo de los puertos UDP. Se realiza mediante el comando "*Nmap*" acompañado de la opción *"--send-ip -sU*", ver en la Figura 3.135.

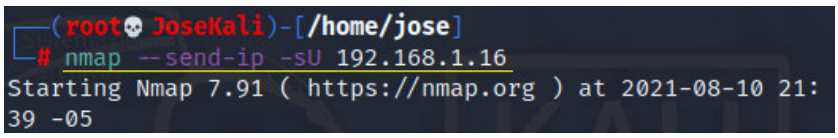


Figura 3.135 Escaneo de Puertos UDP con *Nmap* en Práctica No.2

Honeyd notifica el escaneo de la red en los puertos UDP. Analizando desde la dirección IP del origen con el puerto asignado hacia la dirección IP del destino con puerto correspondiente. El mensaje de interacción de *Honeyd* es "*Connection to closed port*" al español "Conexión a puerto cerrado", ver en la Figura 3.136.

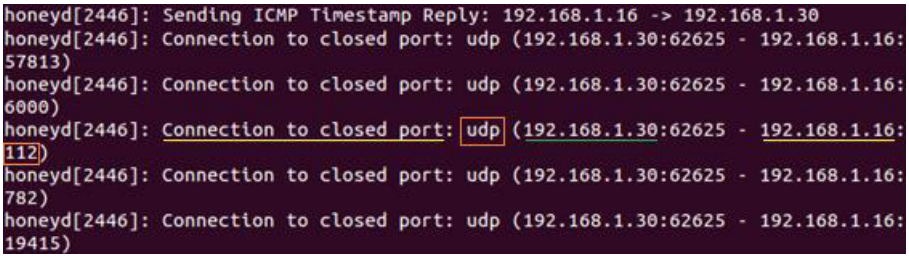


Figura 3.136 Comportamiento de escaneo puertos UDP en Práctica No.2

Cuando finaliza el mapeo, *Nmap* se encarga de ilustrar los resultados de las operaciones realizadas dentro del *Honeypot*. Muestra los puertos UDP otorgados y definidos en el archivo de configuración, ver en la Figura 3.137.

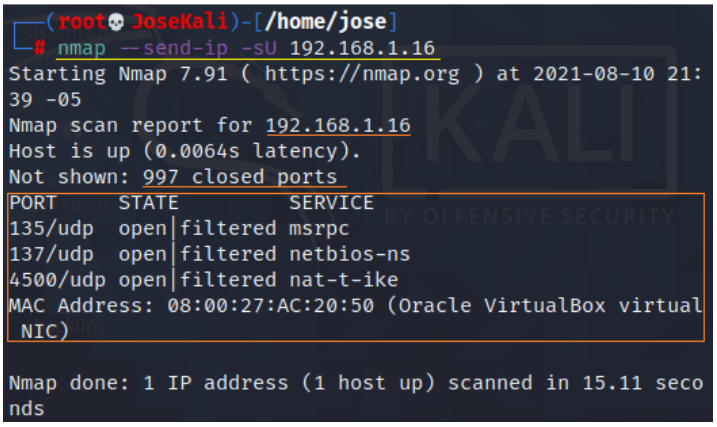


Figura 3.137 Resultados de escaneo puertos UDP en Práctica No.2

Se procede a cerrar el proceso de *Honeyd*, realizando la acción en el teclado “Ctrl+ c”.

Paso 8: Archivo de registro de información

Se realiza el ingreso del archivo de registro mediante el comando “*vim”* acompañado de la ruta *“/var/log/honeyd/* “con el nombre del archivo “*honeyd2.log*”, ver en la Figura 3.138.

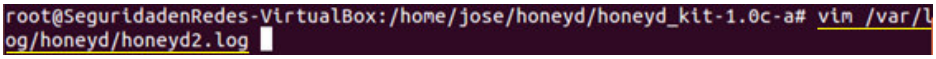


Figura 3.138 Directorio de almacenamiento de información en Práctica No.2

El archivo muestra la información y los datos proporcionados por *Honeyd* a la respuesta del *Honeypot* “*Microsoft Windows XP Professional SP1*”.

A continuación, se presente el análisis de los resultados:

- **Análisis de Mensajes ICMP**

  Interacción de mensajes ICMP entre *host* en las direcciones IP 192.168.1.30 y 192.168.1.16, ver en la Figura 3.139.

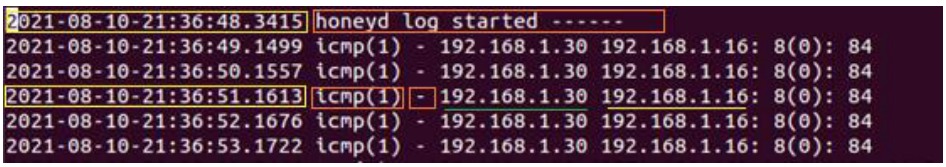


Figura 3.139 Mensajes ICMP en Práctica No.2

1. Fecha y hora de inicio de actividad.
2. Envío de mensaje ICMP entre *host*.

- **Escaneo de puertos TCP**

  Análisis de puertos TCP por medio de *Nmap* en la dirección IP 192.168.1.16, ver en la Figura 3.140.

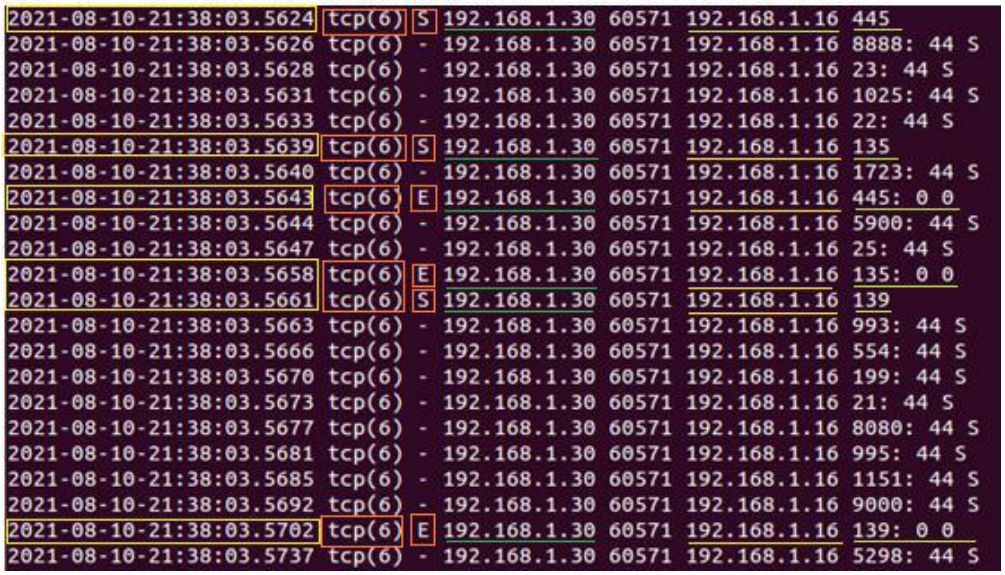


Figura 3.140 Conexiones TCP en Práctica No.2

1. Inicio de conexión (S) en el grupo de puertos TCP 445, 135 y 139.
2. Puertos de origen y destino en direcciones IP.
3. Finalización de conexión (E) en el grupo de puertos TCP 445, 135 y 139.
4. Sin asignación de paquetes por análisis *Nmap.*

- **Escaneo de puertos UDP**

  Análisis de puertos UDP por medio de *Nmap* en la dirección IP 192.168.1.16, ver en la Figura 3.141.

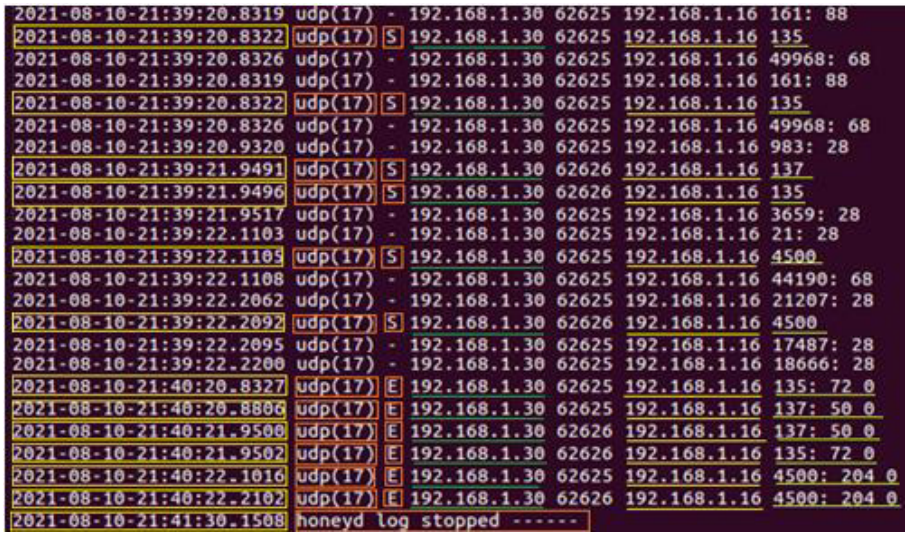


Figura 3.141 Conexiones UDP en Práctica No.2

1. Inicio de conexión (S) en el grupo de puertos UDP 135, 137 y 4500.
2. Puertos de origen y destino en direcciones IP.
3. Finalización de conexión (E) en el grupo de puertos TCP 135, 137 y 4500.
4. Sin asignación de paquetes por análisis *Nmap.*
5. Finalización de proceso de *Honeyd*.

3. CONCLUSIONES

- El archivo de *script* de ARP "*start-arpd.sh*" tiene el objetivo de establecer la comunicación del *Honeypot* creado sobre el sistema *Ubuntu* y permite la validación de conexión con el comando "*ping*" sobre el sistema *Kali Linux*.
- El sistema operativo *Kali Linux* permite escanear y diagnosticar los elementos sustanciales del *Honeypot* mediante el comando "*Nmap*". Por medio del cual, se obtiene la información relacionada a la latencia, número de puertos, estado y servicio al cual se encuentra asignado cada puerto configurado.
- Dentro del sistema operativo *Kali Linux,* los puertos TCP 62625 y UDP 62625 son los encargados de garantizar la entrega de paquetes sobre los puertos TCP y UDP configurados en el equipo *Honeypot* y se muestran en el archivo de registro.

- El archivo "*honeyd2.log*" tiene el objetivo de almacenar la información de la interacción del *Honeypot* creado con el equipo atacante del sistema *Kali Linux*.

4. RECOMENDACIONES

- Se recomienda establecer la ruta del directorio de cada archivo en la ejecución del comando de *Honeyd* con los argumentos, con el fin de proporcionar más información a la terminal del equipo *Ubuntu*.
- Se recomienda realizar el escaneo de puertos una vez que se haya finalizado o interrumpido los mensajes ICMP entre *hosts*, debido a que *Nmap* requiere de un tiempo para obtener los paquetes del equipo *Honeypot*.
- Se recomienda validar las notificaciones de *Honeyd* en función del mensaje de "*Killing attempted connection*" en el escaneo de los puertos TCP y el mensaje de "*Connection to closed port*" en el escaneo de los puertos UDP, para evitar posibles errores de ejecución.
- Se recomienda visualizar con detalle el archivo de registro de información, debido a que algunos paquetes se encuentran ubicados en espacios alejados de unos y de otro, por lo que se necesita visualizar al detalle el archivo.

PRÁCTICA 3

1. **TEMA:** Acciones de Protocolos TCP, UDP e ICMP
2. **DESARROLLO DE LA PRÁCTICA**

NOTA: El tiempo para realizar la práctica es de 30 minutos. Se requiere que el instructor explique los lineamientos de la Red *Honeypot*. Además, es importante que el estudiante realice el trabajo preparatorio de acuerdo con los lineamientos solicitados.

DISEÑO DE RED

El diseño de red se realiza en base a la Red *Honeypot* en la dirección 192.168.1.0/24. El equipo principal utilizado es *Ubuntu*, el cual mediante la herramienta *Honeyd* se procede a emular un conjunto de *Honeypots*. El conjunto se configura en base a 3 personalidades con el objetivo de ejecutar las acciones de protocolos TCP, UDP e ICMP. Asimismo, cuenta con un equipo verificador o atacante denominado *Kali Linux.* La topología de red se visualiza en la Figura 3.142.

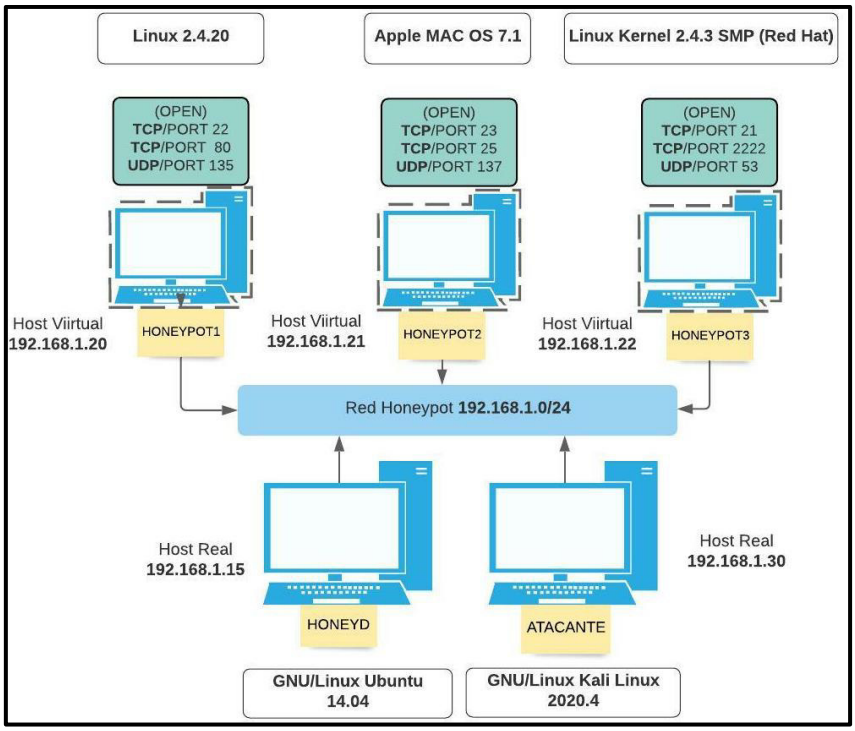


Figura 3.142 Topología de Práctica No.3

Nota: Tomar en cuenta las mismas consideraciones en cuanto a la tarjeta de red, ingreso de máquinas virtuales y manejo de terminal detallado en prácticas anteriores**.**

Paso 1: Elaboración del archivo de configuración con la descripción de los pasos, ver en la Figura 3.143.

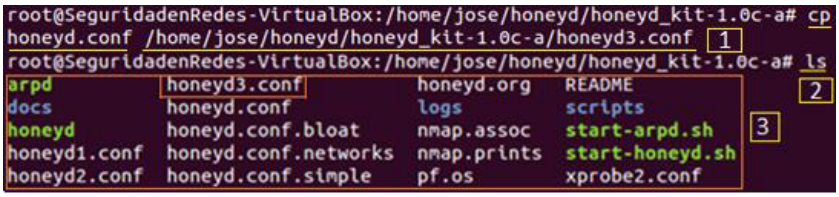


Figura 3.143 Elaboración de archivo de configuración "*honeyd3.conf*"

1. Se procede a realizar una copia del archivo "*honeyd.conf*" con el nombre "*honeyd3.conf*" mediante el comando "*cp*" y la ruta del directorio principal de *Honeyd*.
2. Visualización de archivo creado mediante el comando "*ls*".
3. El archivo "*honeyd3.conf*" en listado de archivos de *Honeyd*.

Paso 2: Selección de los sistemas operativos.

Se procede a seleccionar las personalidades del conjunto de *Honeypots* a emular, ver en la Figura 3.144.

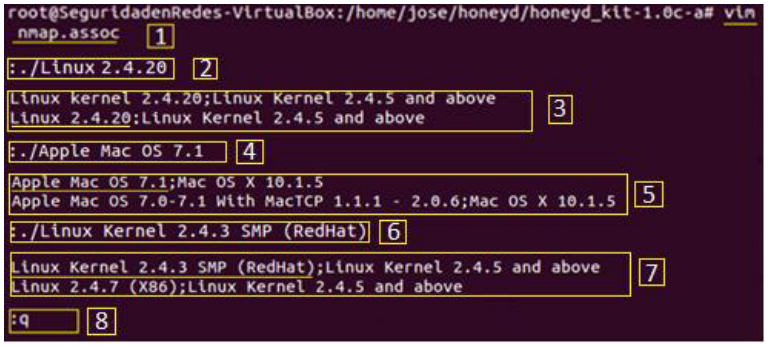


Figura 3.144 Selección de personalidades de Práctica No.3

1. Ingreso al archivo de base de datos de personalidades mediante el comando "*vim*" y nombre del archivo "*Nmap. assoc*".
2. Búsqueda de personalidad mediante "Esc" y el argumento ":./*Linux 2.4.20*".
3. Listado de personalidades y selección de la personalidad indicada.
4. Búsqueda de personalidad con "Esc" y el argumento ":./*Apple MAC OS 7.1*".

5. Listado de personalidades y selección de la personalidad indicada.
6. Búsqueda de personalidad mediante “Esc” y el argumento “:./”*Linux Kernel 2.4.3 SMP (RedHat*)”.
7. Listado de personalidades y selección de la personalidad indicada.
8. Salida de archivo mediante “Esc + q”.

Paso 3: Ingreso al archivo de configuración “*honeyd3.conf*”

Ingreso al archivo de configuración “*honeyd3.conf*” mediante el comando “*vim*”, ver en la Figura 3.145.

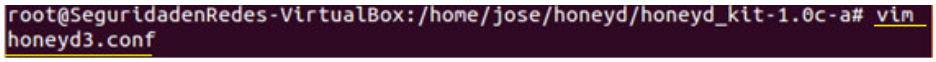


Figura 3.145 Ingreso al archivo “*honeyd3.conf*”

Paso 4: Análisis de acciones de protocolos

Se establece las acciones de protocolos de la Red *Honeypot*. *Honeyd* se encarga de disponer de acciones en cada equipo virtual, ver en disposición de acciones en la Tabla 3.11.

Tabla 3.11 Acciones de protocolos de Práctica No.3 [27] [32]

TCP	
Acción	**Respuesta**
Open	Responde con *Syn/Ack* de conexión (Abierto por defecto)
Block	Descarta el paquete y no responde
Reset	Responde con *RST* (Elimina cualquier intento de conexión)
UDP	
Acción	**Respuesta**
Open	No responde (Cerrado por defecto)
Block	Descarta el paquete y no responde
Reset	Responde con mensaje ICMP de error de puerto
Puerto ICMP	
Acción	**Respuesta**
Open	Responde a los paquetes ICMP (Abierto por defecto)
Block	Descarta el paquete y no responde

Paso 5: Configuración del archivo “*honeyd3.conf*”

El establecimiento de la configuración se realiza mediante el diseño de red y la plantilla por defecto en el apartado final del archivo, ver en la Figura 3.146.

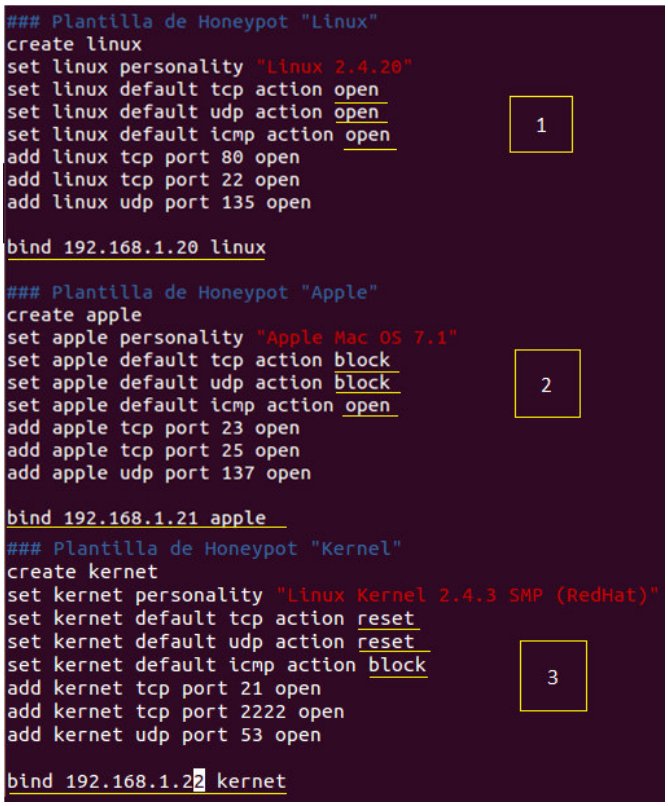


Figura 3.146 Configuración de archivo "*honeyd3.conf*"

Personalidades

Honeypot 1 personalidad "*Linux 2.4.20*"

- Asignación de acción de protocolos TCP y UDP por opción "*open*".
- Habilitación de mensajes ICMP por opción "*open*".
- Asignación de puertos TCP abiertos: 80 y 22.
- Asignación de puerto UDP abierto: 135.
- Designación de dirección IP: 192.168.1.20 en "*linux*".

Honeypot 2 personalidad "*Apple MAC OS 7.1*"

- Asignación de acción de protocolos TCP y UDP por opción "*block*".
- Habilitación de mensajes ICMP por opción "*open*".
- Asignación de puertos TCP abiertos: 23 y 25.
- Asignación de puerto UDP abierto: 137.
- Designación de dirección IP: 192.168.1.21 en "*apple*".

Honeypot 3 personalidad "*Linux Kernel 2.4.3 SMP (RedHat)"*

- Asignación de acción de protocolos TCP y UDP por opción "*reset*".
- Habilitación de mensajes ICMP por opción "*open*".
- Asignación de puertos TCP abiertos: 21 y 2222.
- Asignación de puerto UDP abierto: 53.
- Designación de dirección IP: 192.168.1.22 en "*kernet*".

Paso 6: Ejecución de servicios de inicialización

Se procede a ejecutar los archivos *scripts* de los servicios ARP y *Honeyd*:

- *"./start-arpd.sh*
- "*./start-honeyd.sh*"

Nota: Tomar en consideración el procedimiento de ejecución de los *scripts* ARP y *Honeyd,* realizado en prácticas anteriores.

Paso 7: Inicio de proceso de *Honeyd*

La inicialización se realiza mediante el comando "*honeyd*" acompañado de los argumentos y opciones, ver en la Tabla 3.12.

Tabla 3.12 Comandos de inicialización de *Honeyd* en Práctica No.3 [27] [32]

Opción	Función
honeyd	Arranque de *Honeyd*
-d	Levantamiento del Demonio
-i	Tarjeta de Red "eth0"
-f	Archivo "*Honeyd.conf3*"- ruta
-p	Archivo "*Nmap.prints*" conjunto de personalidades-ruta
-x	Archivo "*xprobe.conf*" sonda-lector de personalidades-ruta
-a	Archivo "*Nmap.assoc*" personalidades-sonda-ruta
-0	Archivo "*pf.os*" lector de personalidades pasivas-ruta
-l	Archivo de Registro de Información
192.168.1.20-192.168.1.22	Rango de direcciones IP de *Honeypots.*

Se realiza la implementación del comando con los argumentos, ver en la Figura 3.147.

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd/honeyd_kit-1.0c-a# honeyd -d -i
eth0 -f /home/jose/honeyd/honeyd_kit-1.0c-a/honeyd3.conf -p /home/jose/honeyd/honey
d_kit-1.0c-a/nmap.prints -x /home/jose/honeyd/honeyd_kit-1.0c-a/xprobe2.conf -a /ho
me/jose/honeyd/honeyd_kit-1.0c-a/nmap.assoc -0 /home/jose/honeyd/honeyd_kit-1.0c-a/
pf.os 192.168.1.20-192.168.1.22 -l /tmp/honeyd3.log
```

Figura 3.147 Argumentos de inicialización de *Honeyd* en Práctica No.3

Nota: Antes de realizar la acción de *Honeyd*, es importante tener habilitado el sistema de *Kali Linux* para mejorar la experiencia y eficacia de la práctica.

Honeyd muestra los argumentos, advertencias y procesos del servicio, ver en la Figura 3.148.

```
Honeyd V1.5c Copyright (c) 2002-2007 Niels Provos
honeyd[2861]: started with -d -i eth0 -f /home/jose/honeyd/honeyd_kit-1.0c-a/hon
eyd3.conf -p /home/jose/honeyd/honeyd_kit-1.0c-a/nmap.prints -x /home/jose/honey
d/honeyd_kit-1.0c-a/xprobe2.conf -a /home/jose/honeyd/honeyd_kit-1.0c-a/nmap.ass
oc -0 /home/jose/honeyd/honeyd_kit-1.0c-a/pf.os -l /var/log/honeyd/honeyd3.log 1
92.168.1.20-192.168.1.22
Warning: Impossible SI range in Class fingerprint "IBM OS/400 V4R2M0"
Warning: Impossible SI range in Class fingerprint "Microsoft Windows NT 4.0 SP3"
honeyd[2861]: listening promiscuously on eth0: (arp or ip proto 47 or (udp and s
rc port 67 and dst port 68) or (ip and (dst net 192.168.1.20/31 or dst net 192.1
68.1.22/32))) and not ether src 08:00:27:ac:20:50
honeyd[2861]: Demoting process privileges to uid 65534, gid 65534
```

Figura 3.148 Levantamiento de *Honeyd* en Práctica No.3

Paso 8: Validación de *Honeypot* en *Kali Linux.*

Nota: Validar las conexiones de las tarjetas de red y configuración de la dirección IP del equipo *Kali Linux* de acuerdo a las prácticas anteriores.

Se procede a verificar la Red *Honeypot* iniciando en la máquina "*Linux 2.4.20*" mediante el comando "*ping*" en la dirección IP 192.168.1.20, ver en la Figura 3.149.

```
┌──(root💀JoseKali)-[/home/jose]
└─# ping 192.168.1.20
PING 192.168.1.20 (192.168.1.20) 56(84) bytes of data.
64 bytes from 192.168.1.20: icmp_seq=1 ttl=255 time=1033 ms
64 bytes from 192.168.1.20: icmp_seq=2 ttl=255 time=4.18 ms
64 bytes from 192.168.1.20: icmp_seq=3 ttl=255 time=1.45 ms
^C
--- 192.168.1.20 ping statistics ---
4 packets transmitted, 3 received, 25% packet loss, time 3032ms
rtt min/avg/max/mdev = 1.451/346.346/1033.403/485.823 ms, pipe 2
```

Figura 3.149 Comando "*ping*" sobre *Honeypot* 1 en Práctica No.3

Se procede a cerrar el proceso del comando "*ping*" realizando la acción en el teclado "Ctrl+ c".

El establecimiento de la conexión entre máquinas se realiza con los mensajes ICMP. El proceso de envío de mensajes desde el *Honeypot* 192.168.1.20 hacia el equipo externo 192.168.1.30 se valida con el *host* virtual, ver en la Figura 3.150.

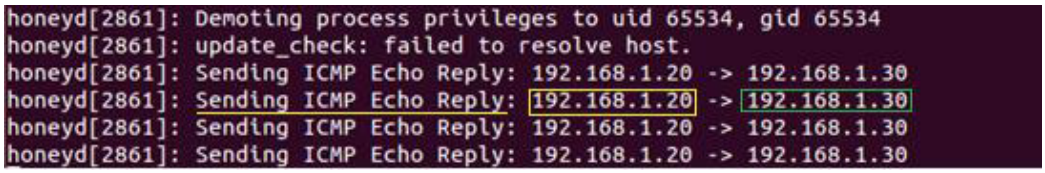


Figura 3.150 Conexión de *Honeypot* 1 con *Kali Linux* en Práctica No.3

En *Kali Linux* se procede a escanear el *host* mediante la herramienta *Nmap*. El comando "*Nmap*" se acompaña de la opción "*--send-ip -P0*" para el análisis de los puertos TCP en la dirección IP 192.168.1.20, ver en la Figura 3.151.

Figura 3.151 Escaneo de puertos TPC en *Honeypot* 1 en Práctica No.3

Honeyd notifica el escaneo de la red en los puertos TCP. Analiza la dirección IP de origen con el puerto asignado hacia la dirección IP de destino con puerto correspondiente. El mensaje de interacción de *Honeyd* es "*Connection dropped by reset*" al español "Conexión Interrumpida por reinicio", ver en la Figura 3.152.

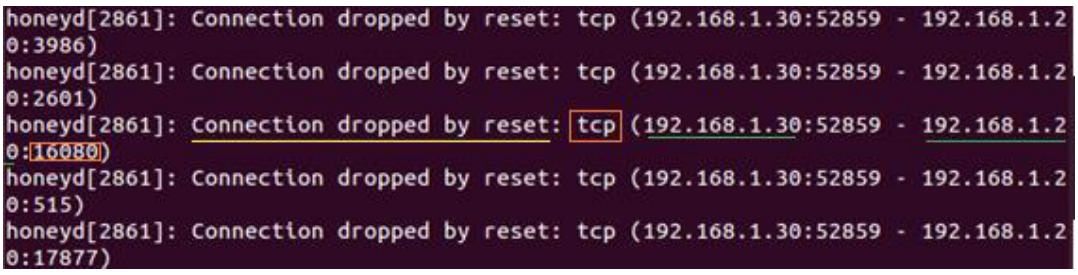


Figura 3.152 Interacción de puertos TCP en *Honeypot* 1 en Práctica No.3

El resultado del escaneo muestra a todos los puertos "abiertos" debido a la configuración de la acción del protocolo TCP en la dirección IP 192.168.1.20 *Honeypot* "*Linux 2.4.20*", ver en las Figura 3.153 y Figura 3.154.

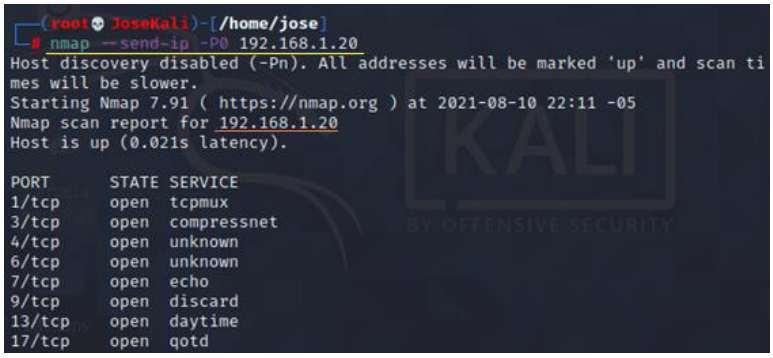


Figura 3.153 Resultado de escaneo TCP en *Honeypot* 1 en Práctica No.3

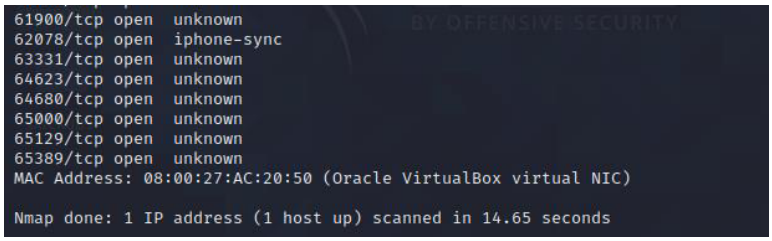


Figura 3.154 Resultado final de escaneo TCP en *Honeypot* 1 en Práctica No.3

Del mismo modo se procede a realizar el escaneo de los puertos UDP. Se realiza mediante el comando "*Nmap*" acompañado de la opción *"--send-ip -sU*" en la dirección IP 192.168.1.20,ver en la Figura 3.155.

Figura 3.155 Escaneo de puertos UDP en *Honeypot* 1 en Práctica No.3

Honeyd notifica el escaneo de la red en los puertos UDP. El análisis muestra el establecimiento de conexión de cada uno de los puertos, ver en la Figura 3.156.

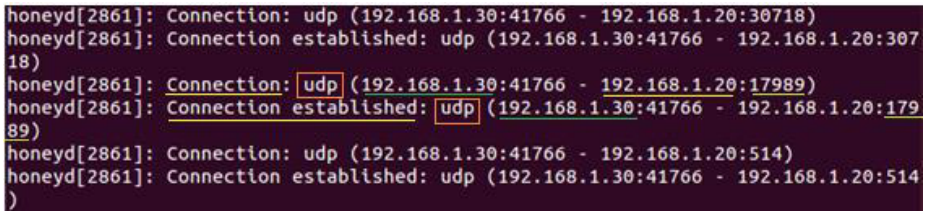


Figura 3.156 Interacción de puertos UDP en *Honeypot* 1 en Práctica No.3

El resultado del escaneo determina que todos los puertos se encuentran abiertos en la dirección 192.168.1.20. Debido a la acción "*open*" en el protocolo UDP, ver en la Figura 3.157.

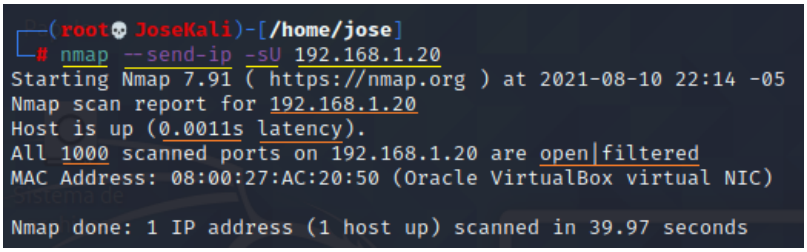


Figura 3.157 Resultado de escaneo UDP en *Honeypot* 1 en Práctica No.3

Continuando, se procede con la máquina *Honeypot* "*Apple MAC OS 7.1*" mediante el comando "*ping*" en la dirección IP 192.168.1.21, ver en la Figura 3.158.

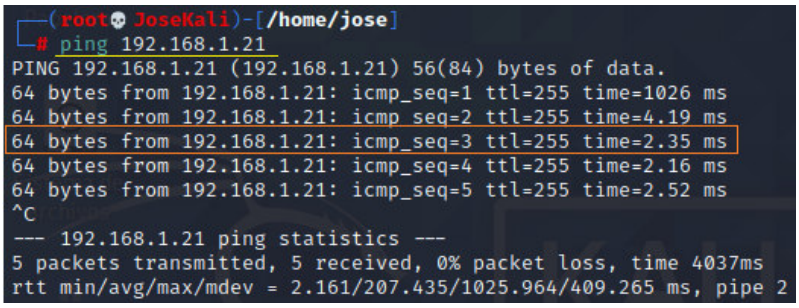


Figura 3.158 Comando "*ping*" sobre *Honeypot* 2 en Práctica No.3

Establecimiento de conexión entre máquinas por medio de mensajes ICMP. El proceso de envío de mensajes desde el *Honeypot* 192.168.1.21 hacia el equipo externo 192.168.1.30 se valida con el *host* virtual, ver en la Figura 3.159.

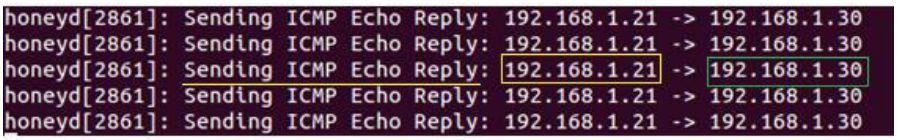


Figura 3.159 Conexión de *Honeypot* 2 con *Kali Linux* en Práctica No.3

En *Kali Linux* se procede a escanear el *host* mediante la herramienta *Nmap*. El comando "*Nmap*" se acompaña de la opción "*--send-ip -P0*" para el análisis de los puertos TCP en la dirección IP 192.168.1.21, ver en la Figura 3.160.

Figura 3.160 Escaneo de puertos TCP en *Honeypot* 2 en Práctica No.3

Honeyd notifica el establecimiento de la conexión única y exclusivamente con los puertos 23 y 25. ver en la Figura 3.161.

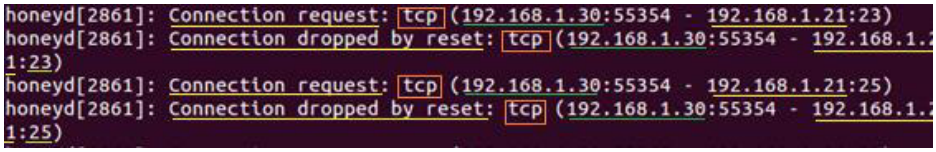


Figura 3.161 Interacción de puertos TCP en *Honeypot* 2 en Práctica No.3

El escaneo muestra puertos que se establecieron como abiertos y determinada a los demás como filtrados por la acción de bloqueo "*block*", ver en la Figura 3.162.

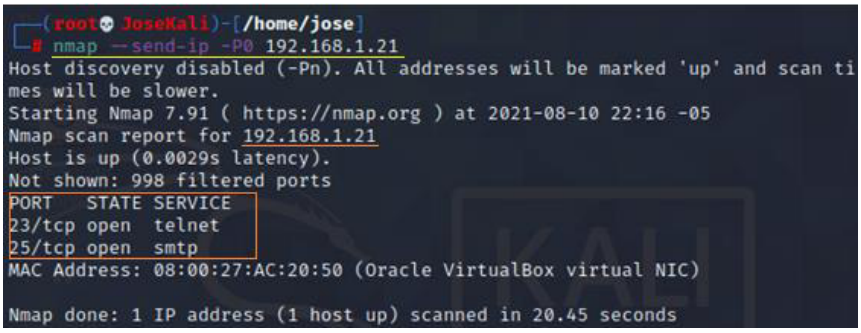


Figura 3.162 Resultado de escaneo TCP en *Honeypot* 2 en Práctica No.3

Del mismo modo se procede a realizar el escaneo de los puertos UDP. Se realiza mediante el comando "*Nmap*" acompañado de la opción *"--send-ip -sU*" en la dirección IP 192.168.1.21, ver en la Figura 3.163.

Figura 3.163 Escaneo de puertos UDP en *Honeypot* 2 en Práctica No.3

Honeyd notifica el establecimiento de conexión del puerto UDP 137, configurado previamente en el archivo configuración, ver en la Figura 3.164.

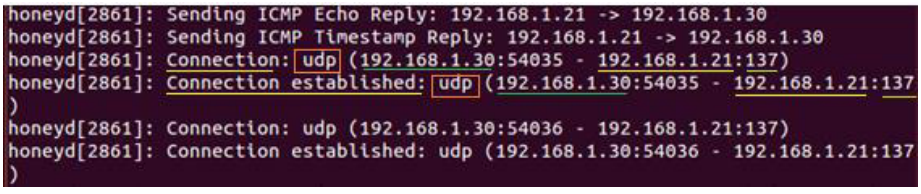


Figura 3.164 Interacción de puertos UDP en *Honeypot* 2 en Práctica No.3

El resultado del escaneo determina que todos los puertos se encuentran abiertos. Sin embargo, debido a que *Honeyd* establece a un único puerto habilitado decreta a los demás puertos como disponibles. En *Nmap* se notifica el estado habilitado en la

dirección IP "192.168.1.21". Es el resultado de la acción "*block*" en el protocolo UDP, ver en la Figura 3.165.

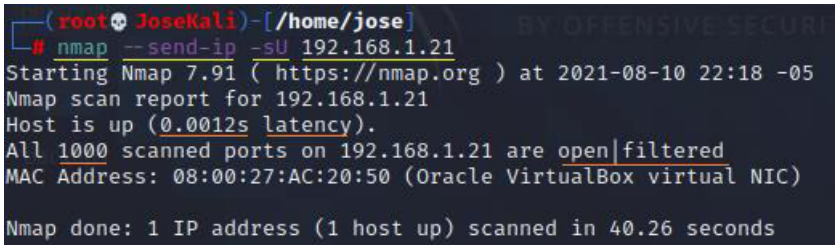


Figura 3.165 Resultado de escaneo UDP en *Honeypot* 2 en Práctica No.3

Al final, se procede a validar la máquina *Honeypot* "*Linux Kernel 2.4.3 SMP (RedHat)"* mediante el comando "*ping*" en la dirección IP 192.168.1.22, ver en la Figura 3.166.

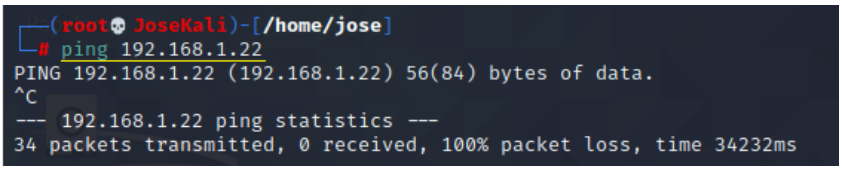


Figura 3.166 Comando "*ping*" sobre *Honeypot* 3 en Práctica No.3

Se justica la inexistencia de los mensajes ICMP por medio del archivo de configuración del *Honeypot*, debido al establecimiento de la acción "*block*" en el protocolo ICMP. Razón por la cual se produce un bloqueo en todos los mensajes ICMP entre los *hosts*. Lo que imposibilita la comunicación por medio del comando "*ping*".

En *Kali Linux* se procede a escanear el *host* mediante la herramienta *Nmap*. El comando "*Nmap*" se acompaña de la opción "*--send-ip -P0*" para el análisis de los puertos TCP en la dirección IP 192.168.1.22, ver en la Figura 3.167.

Figura 3.167 Escaneo de puertos TCP en *Honeypot* 3 en Práctica No.3

En *Honeyd* se notifica mediante el mensaje "*Killing attempted connection*" en español "Intento de conexión eliminado", debido a que se realiza el escaneo y validación en cada puerto por medio del comando "*Nmap*", ver en la Figura 3.168.

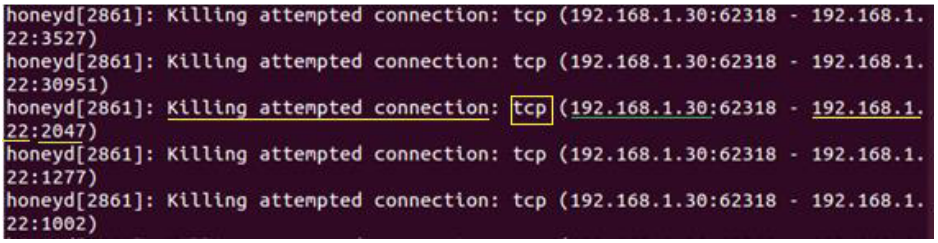


Figura 3.168 Interacción de puertos TCP en *Honeypot* 3 en Práctica No.3

El escaneo mediante el comando "*Nmap*" muestra los puertos que se establecieron como abiertos y determinada a los demás por cerrados debido a la acción de *reseteo* "*reset*" configurado previamente en archivo de configuración, ver en la Figura 3.169.

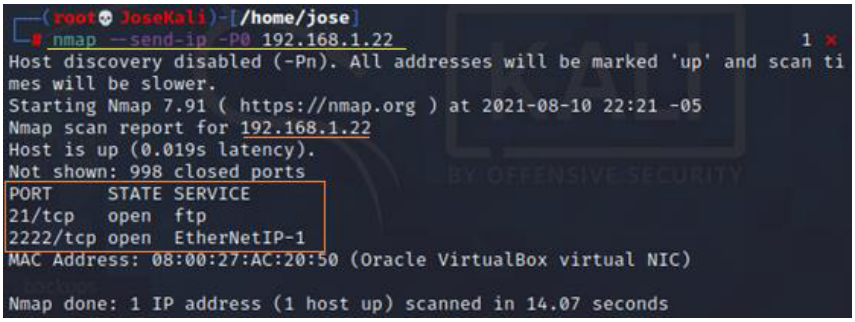


Figura 3.169 Resultado de escaneo TCP en *Honeypot* 3 en Práctica No.3

Del mismo modo se procede a realizar el escaneo de los puertos UDP. Se realiza mediante el comando "*Nmap*" acompañado de la opción *"--send-ip -sU*" en la dirección IP 192.168.1.22, ver en la Figura 3.170.

Figura 3.170 Escaneo de puertos UDP en *Honeypot* 3 en Práctica No.3

Honeyd notifica el mensaje "*Connection to closed port*" al español "Conexión a puerto cerrado". Evidenciando el cierre de cada uno de los puertos en el análisis, ver en la Figura 3.171

Figura 3.171 Interacción de puertos UDP en *Honeypot* 3 en Práctica No.3

El resultado del escaneo determina al puerto UDP 53 como abierto y los demás puertos cerrados, tal como se configuró en el archivo de configuración, ver en la Figura 3.172.

Nota: La acción "*reset*" se recomienda por parte de *Honeyd* para realizar las configuraciones de los archivos de ejecución.

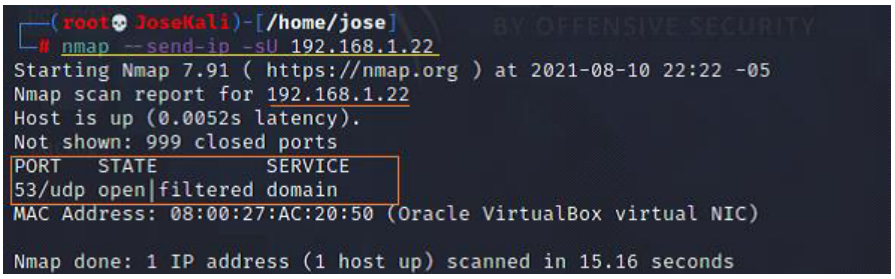


Figura 3.172 Resultado de escaneo UDP en *Honeypot* 3 en Práctica No.3

Se procede a cerrar el proceso de *Honeyd* mediante "Ctrl+ c".

Paso 9: Archivo de registro de información

Se realiza el ingreso del archivo de registro mediante el comando "*vim*" acompañado de la ruta *"/var/log/honeyd/* "con el nombre del archivo "*honeyd3.log*", ver en la Figura 3.173.

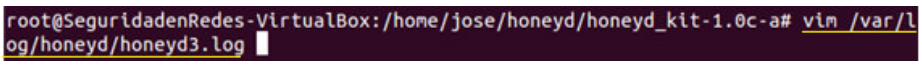


Figura 3.173 Archivo de registro en Práctica No.3

El archivo muestra el análisis y los datos proporcionados por *Honeyd* a la respuesta de la Red *Honeypot*. A continuación, se detallan los datos registrados en cada una de las personalidades:

- **Análisis *Honeypot* "*Linux 2.4.20*".**
  El análisis de mensajes ICMP y puertos TCP se visualiza en la Figura 3.174.

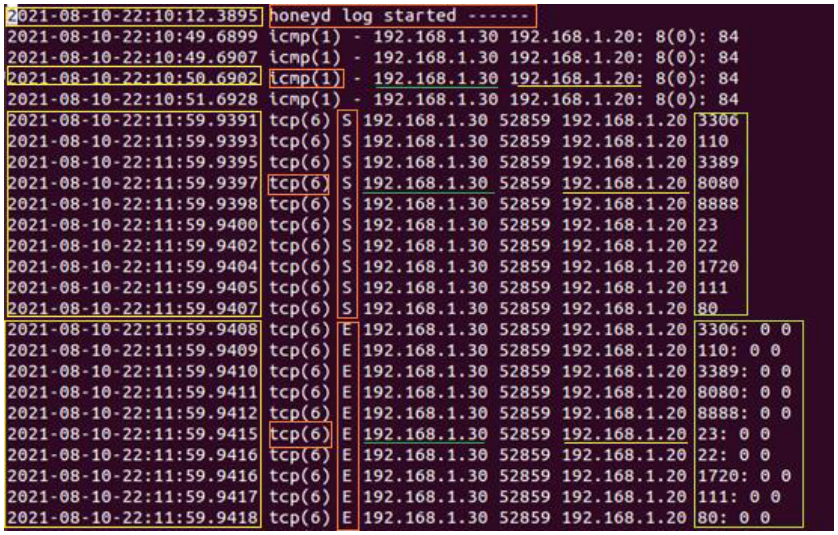


Figura 3.174 Análisis ICMP y TCP en *Honeypot* 1 en Práctica No.3

1. Fecha y hora de Inicialización del proceso de *Honeyd.*
2. Mensaje “ICMP” enviado entre *host*.
3. Análisis de puertos TCP.
4. Inicio de conexión (S) en grupo de puertos TCP.
5. Finalización de conexión (E) en grupo de puertos TCP.

Nota: La acción “*open*” establece la conexión de cada uno de puertos por lo que *Honeyd* toma grupo de puertos, inicia y termina la conexión en los mismos.

El análisis de los puertos UDP se visualiza en la Figura 3.175.

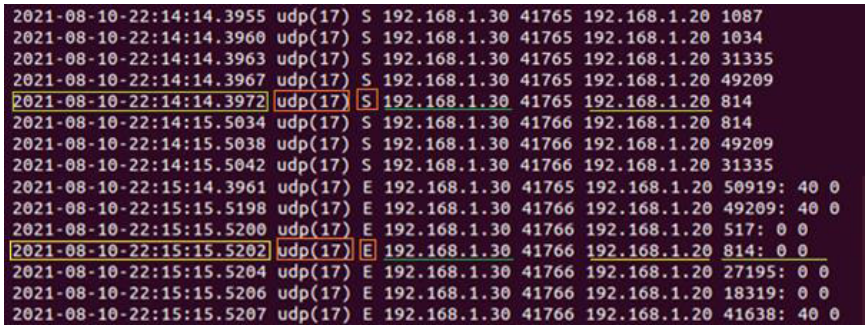


Figura 3.175 Análisis UDP en *Honeypot* 1 en Práctica No.3

1. Inicio de conexión (S) de puertos UDP.
2. Finalización de conexión (E) de puertos UDP.

Nota: La acción "*open*" en los puertos UDP establece la conexión inicial con cada uno de los puertos y una vez terminado finaliza la misma conexión.

- **Análisis *Honeypot* "*Apple MAC 7.1*"**

  El análisis de mensajes ICMP y puertos TCP se visualiza en la Figura 3.176.

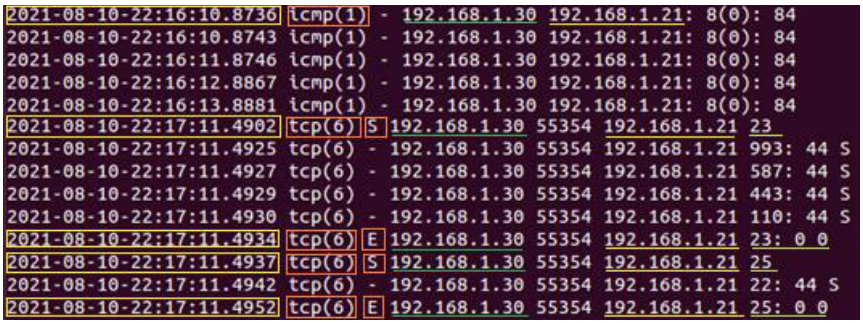


Figura 3.176 Análisis TCP en *Honeypot* 2 en Práctica No.3

1. Fecha y hora de proceso de *Honeyd.*
2. Mensaje "ICMP" enviado entre *host*.
3. Análisis de puertos TCP.
4. Inicio de conexión (S) en grupo de puertos TCP 23 y 25.
5. Finalización de conexión (E) en grupo de puertos TCP 23 y 25.

Nota: La acción "*block*" estable la conexión exclusiva en el puerto configurado en el archivo.

El análisis UDP de los puertos se visualiza en la Figura 3.177.

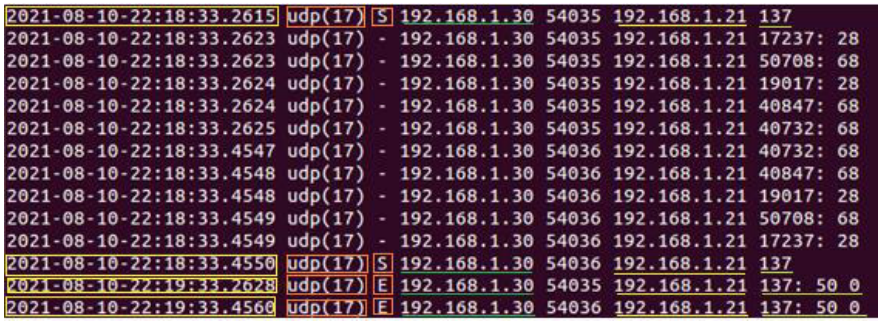


Figura 3.177 Análisis UDP en *Honeypot* 2 en Práctica No.3

1. Inicio de conexión (S) puerto UDP 137.
2. Finalización de conexión (E) puerto UDP 137.

Nota: La acción "*block*" en los puertos UDP descarta los paquetes de los puertos y establece la conexión con el puerto configurado en el archivo.

- **Análisis *Honeypot* "*Linux Kernel 2.4.3 SMP (RedHat)*"**

  El análisis de mensajes ICMP y puertos TCP se visualiza en la Figura 3.178.

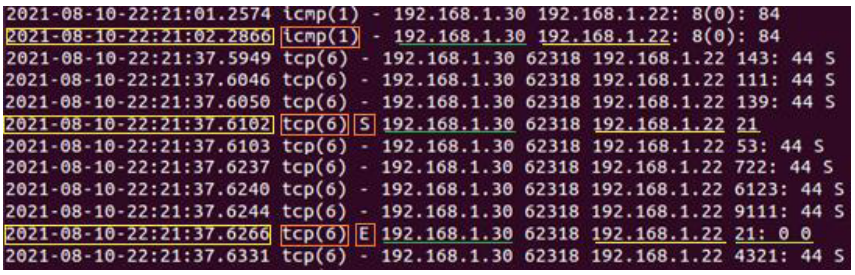


Figura 3.178 Análisis TCP en *Honeypot* 3 en Práctica No.3

1. Fecha y hora de proceso de *Honeyd.*
2. Sin mensaje "ICMP".
3. Análisis de puertos TCP.
4. Inicio de conexión (S) en puerto TCP 21.
5. Finalización de conexión (E) en puerto TCP 21.

Nota: La acción "*block*" no permite el envío de mensajes ICMP entre *host*. Además, la acción "reset" establece la conexión exclusiva en el puerto configurado en el archivo como abierto.

El análisis TCP de los puertos se visualiza en la Figura 3.179.

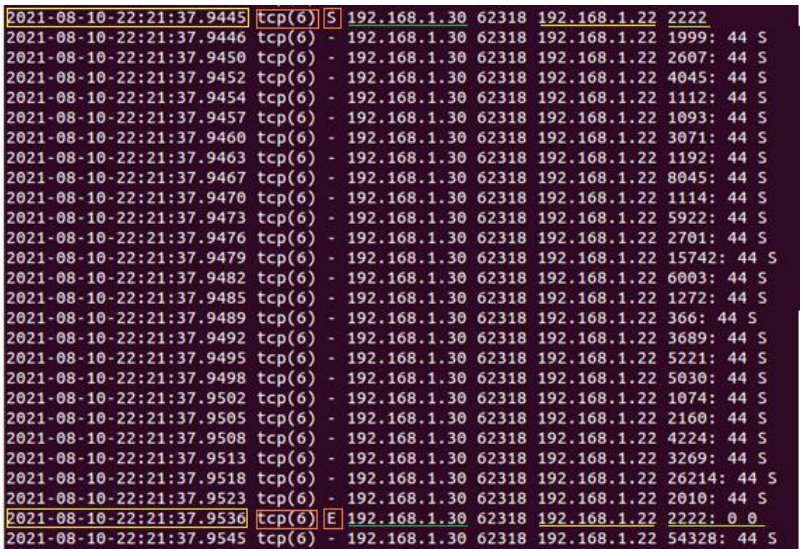


Figura 3.179 Análisis amplio TCP en *Honeypot* 3 en Práctica No.3

1. Inicio de conexión (S) en puerto TCP 2222.
2. Finalización de conexión (E) en puerto TCP 2222.

Se realiza en análisis de los puertos UDP, ver en la Figura 3.180.

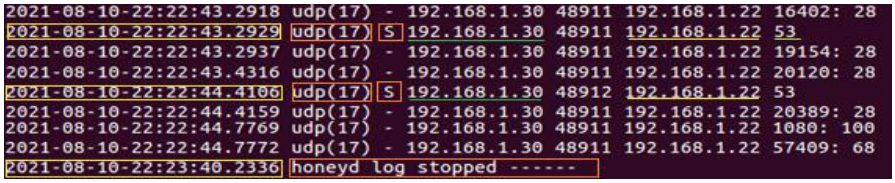


Figura 3.180 Análisis UDP en *Honeypot* 3 en Práctica No.3

1. Inicio de conexión (S) puerto UDP 53.
2. Finalización de conexión (E) puerto UDP 53.
3. Finalización del proceso de *Honeyd*.

Nota: La acción "*reset*" en los puertos UDP envía mensajes de error ICMP y establece la conexión en puertos configurados en el archivo.

3. CONCLUSIONES

- La acción "*open*" permite configurar a todos los puertos TCP o UDP en el estado abierto, de tal manera que el atacante encuentra más atractivo al *Honeypot* dentro de la red.
- La acción "*block*" protege a los puertos TCP o UDP; es decir, mediante su configuración, el atacante no obtiene visibilidad, asegurando el estado del puerto.
- La acción "*reset*" utiliza la configuración de conexión en puertos TCP o UDP para asegurar y establecer configuraciones más seguras y específicas.

4. RECOMENDACIONES

- Se recomienda ajustar las acciones "*open*", "*block*" y "*reset*" por defecto de *Honeyd* a las personalidades de acuerdo al orden de interacción y validar de manera coordinada en el equipo *Kali Linux.*
- Si se detecta en el análisis de la personalidad "*Apple MAC 7.1*" en la acción por "*block*" el estado de habitación de puertos abiertos, se recomienda validar el estado del único del puerto abierto en la notificación de *Honeyd*, con el fin de evitar algún tipo de error y certificar el estado.
- Se sugiere invertir el tiempo necesario en el archivo de registro para evitar cualquier tipo de error y encontrar la información necesaria del análisis *Nmap*.

PRÁCTICA 4

1. **TEMA:** Simulación de Servicios TCP
2. **DESARROLLO DE LA PRÁCTICA**

NOTA: El tiempo para realizar la práctica es de 40 minutos. Se requiere que el instructor explique los lineamientos de la Red *Honeypot*. Además, es importante que el estudiante realice el trabajo preparatorio de acuerdo con los lineamientos solicitados.

DISEÑO DE RED

El diseño de red se realiza en base a la Red *Honeypot* en la dirección 192.168.1.0/24. El equipo principal utilizado es *Ubuntu*, el cual mediante la herramienta *Honeyd* se procede a emular un conjunto de *Honeypots*. El conjunto se configura en base a 3 personalidades con el objetivo de validar la simulación de servicios TCP. Asimismo, cuenta con un equipo verificador o atacante denominado *Kali Linux*. La topología de red se visualiza en la Figura 3.181.

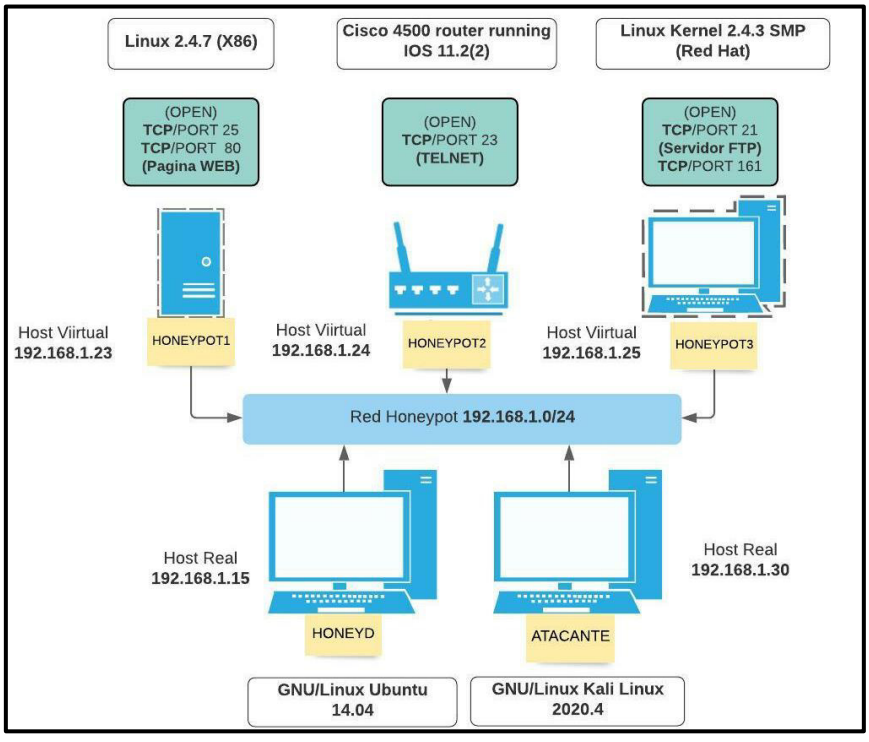


Figura 3.181 Topología de Práctica No.4

Nota: Tomar en cuenta las mismas consideraciones en cuanto a tarjeta de red, ingreso de máquinas virtuales y manejo de terminal detallado en prácticas anteriores**.**

Paso 1: Elaboración del archivo de configuración con la descripción de los pasos, ver en la Figura 3.182.

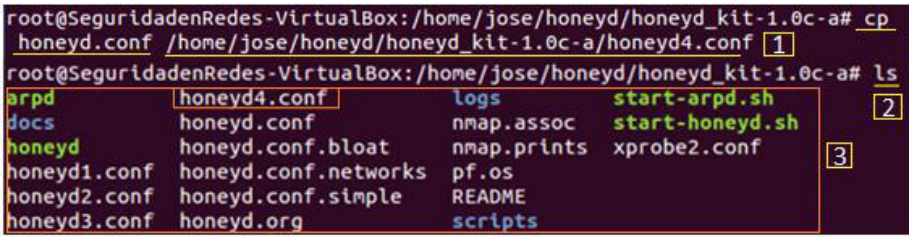


Figura 3.182 Elaboración de archivo de configuración "*honeyd4.conf*"

1. Se procede a realizar una copia del archivo "*honeyd.conf*" con el nombre "*honeyd4.conf*" mediante el comando "*cp*" y la ruta del directorio principal de *Honeyd.*
2. Visualización de archivo creado mediante el comando "*ls*".
3. El archivo "*honeyd4.conf*" en listado de archivos de *Honeyd*.

Paso 2:Selección de los sistemas operativos.

Se procede a seleccionar las personalidades del conjunto de *Honeypots* a emular, ver en la Figura 3.183.

Figura 3.183 Selección de personalidades de Práctica No.4

1. Ingreso al archivo de base de datos de personalidades mediante el comando "*vim*" y nombre del archivo "*Nmap. assoc*".
2. Búsqueda de personalidad mediante "Esc" y el argumento *":./Linux 2.4.7*".
3. Listado de personalidades y selección de la personalidad indicada.

4. Búsqueda de personalidad mediante “Esc” y el argumento “:*./Cisco 4500 router running IOS 11.2(2)”.*
5. Listado de personalidades y selección de la personalidad indicada.
6. Búsqueda de personalidad mediante “Esc” y el argumento “:./”*Linux Kernel 2.4.3 SMP (RedHat)*”.
7. Listado de personalidades y selección de la personalidad indicada.
8. Salida de archivo mediante “Esc + q”.

Paso 3: Ingreso al archivo de configuración “*honeyd4.conf*”

Ingreso al archivo de configuración “*honeyd4.conf*” mediante el comando “*vim*”, ver en la Figura 3.184.

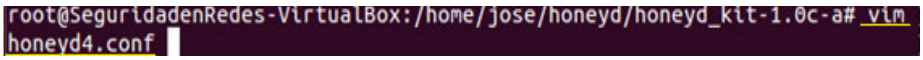


Figura 3.184 Ingreso del archivo “*honeyd4.conf*”

Paso 4: Líneas de comando de archivo de configuración.

Se destalla las líneas de comando del archivo para el establecimiento de servicios, se describe en la Tabla 3.13.

Tabla 3.13 Líneas de comando en configuración de Práctica No.4 [27] [32]

Argumento	Función
create	Creación de Personalidad o Sistema Operativo
set	Establecer acción
default	Por defecto
add	Agregar función o puerto
TCP/UDP	Protocolos
port	Puerto
open	Abierto
/scripts/	Directorio de *scripts* de simulación
$ipsrc $sport	Variables de dirección IP fuente - puerto fuente
$ipdst $dport	Variables de dirección IP destino - puerto destino
perl	Archivo tipo “.pl”
uptime	Tiempo de subida en segundos
bind	Enlace o Dirección IP

Paso 5: Configuración del archivo "*honeyd4.conf*"

El establecimiento de la configuración se realiza mediante el diseño de red y la plantilla por defecto en el apartado final del archivo, ver en la Figura 3.185.

Figura 3.185 Configuración del archivo "*Honeyd4.conf*"

Personalidades

Honeypot 1 personalidad "*Linux 2.4.7 (X86*)"

- Asignación de acción de TCP y UDP por opciones de "*reset*" y "*block*".
- Habilitación de mensajes ICMP por opción "*open*".
- Establecimiento "*uptime*" en tiempo de subida 79239 (ms).
- Asignación de puerto TCP 80 para servicio de página Web (Apache).
- Ruta de archivos de servicio.
- Asignación de puerto TCP abierto: 25.
- Designación de dirección IP: 192.168.1.23 en "*suse80*".

Honeypot 2 personalidad "*Cisco 4500 router running IOS 11.2(2)"*

- Asignación de acción de TCP y UDP por opciones de "*reset*" y "*block*".
- Habilitación de mensajes ICMP por opción "*open*".
- Establecimiento "*uptime*" en tiempo de subida 79239 (ms).
- Asignación de puerto TCP 23 para servidor Telnet en *router*.
- Ruta de archivos de servicio.
- Designación de dirección IP: 192.168.1.24 en "*router*".

Honeypot 3 personalidad "*Linux Kernel 2.4.3 SMP (RedHat)"*

- Asignación de acción de TCP y UDP por opciones de "*reset*" y "*block*".
- Habilitación de mensajes ICMP por opción "*open*".
- Establecimiento "*uptime*" en tiempo de subida 79239 (ms).
- Asignación de puerto TCP 21 para servidor FTP.
- Ruta de archivos de servicio.
- Asignación de puerto TCP abierto: 161.
- Designación de dirección IP: 192.168.1.25 en "*kernel*".

Paso 6: Ejecución de servicios de inicialización

Se procede a ejecutar los archivos *scripts* de los servicios ARP y *Honeyd*:

- *"./start-arpd.sh*
- "*./start-honeyd.sh*"

Nota: Tomar en consideración el procedimiento de ejecución de los *scripts* ARP y *Honeyd*, realizado en prácticas anteriores.

Paso 7: Inicio del proceso de *Honeyd*

La inicialización se realiza mediante el comando "*honeyd*" acompañado de argumentos y opciones, ver en la Tabla 3.14.

Tabla 3.14 Comandos de inicialización de *Honeyd* en Práctica No.4 [27] [32]

Opción	Función
honeyd	Arranque de *Honeyd*
-d	Levantamiento del Demonio
-i	Tarjeta de Red "eth0"
-f	Archivo "*Honeyd.conf4*"- ruta

-p	Archivo "*Nmap.prints*" conjunto de personalidades-ruta
-x	Archivo "*xprobe.conf*" sonda-lector de personalidades-ruta
-a	Archivo "*Nmap.assoc*" personalidades-sonda-ruta
-0	Archivo "*pf.os*" lector de personalidades pasivas-ruta
-l	Archivo de Registro de Información
192.168.1.23-192.168.1.25	Rango de direcciones IP de *Honeypots.*

Se realiza la implementación del comando con los argumentos, ver en la Figura 3.186.

```
root@SeguridadenRedes-VirtualBox:/home/jose/honeyd/honeyd_kit-1.0c-a# honeyd
 -d -i eth0 -f /home/jose/honeyd/honeyd_kit-1.0c-a/honeyd4.conf -p /home/jos
e/honeyd/honeyd_kit-1.0c-a/nmap.prints -a /home/jose/honeyd/honeyd_kit-1.0c-
a/nmap.assoc -x /home/jose/honeyd/honeyd_kit-1.0c-a/xprobe2.conf -0 /home/jo
se/honeyd/honeyd_kit-1.0c-a/pf.os -l /var/log/honeyd/honeyd4.log 192.168.1.2
3-192.168.1.25
```

Figura 3.186 Argumentos de inicialización de *Honeyd* en Práctica No.4

Nota: Antes de realizar la acción de *Honeyd*, es importante tener habilitado el sistema de *Kali Linux* para mejor la experiencia y eficacia de la práctica.

Honeyd muestra los argumentos, advertencias y procesos del servicio, ver en la Figura 3.187.

```
Honeyd V1.5c Copyright (c) 2002-2007 Niels Provos
honeyd[4996]: started with -d -i eth0 -f /home/jose/honeyd/honeyd_kit-1.0c-a
/honeyd4.conf -p /home/jose/honeyd/honeyd_kit-1.0c-a/nmap.prints -a /home/jo
se/honeyd/honeyd_kit-1.0c-a/nmap.assoc -x /home/jose/honeyd/honeyd_kit-1.0c-
a/xprobe2.conf -0 /home/jose/honeyd/honeyd_kit-1.0c-a/pf.os -l /var/log/hone
yd/honeyd4.log 192.168.1.23-192.168.1.25
Warning: Impossible SI range in Class fingerprint "IBM OS/400 V4R2M0"
Warning: Impossible SI range in Class fingerprint "Microsoft Windows NT 4.0
SP3"
honeyd[4996]: listening promiscuously on eth0: (arp or ip proto 47 or (udp a
nd src port 67 and dst port 68) or (ip and (dst net 192.168.1.23/32 or dst n
et 192.168.1.24/31))) and not ether src 08:00:27:ac:20:50
honeyd[4996]: Demoting process privileges to uid 65534, gid 65534
```

Figura 3.187 Arranque de *Honeyd* en Práctica No.4

Paso 8: Validación de *Honeypot* en *Kali Linux*

Nota: Validar las conexiones de las tarjetas de red y configuración de la dirección IP del equipo *Kali Linux* de acuerdo a las prácticas anteriores.

Se procede a verificar la Red *Honeypot* iniciando en la máquina "*Linux 2.4.7 (X86)*" mediante el comando "*ping"* en la dirección IP 192.168.1.23, ver en la Figura 3.188.

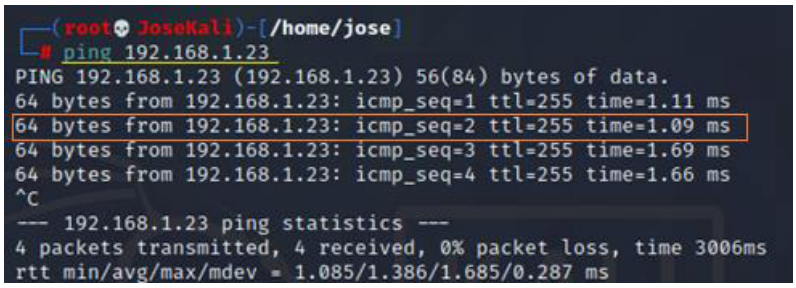


Figura 3.188 Comando "*ping*" en *Honeypot* 1 en Práctica No.4

Se procede a cerrar el proceso del comando "*ping*" realizando la acción en el teclado "Ctrl+ c".

El establecimiento de la conexión entre máquinas se realiza con los mensajes ICMP. El proceso de envío de mensajes desde el *Honeypot* 192.168.1.23 hacia el equipo externo 192.168.1.30 se valida con el *host* virtual, ver en la Figura 3.189.

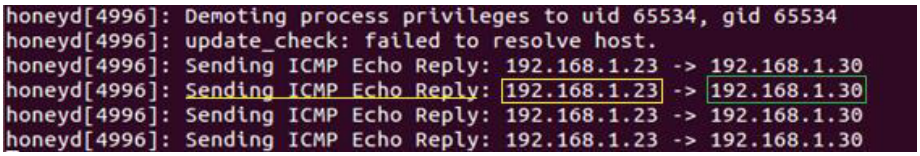


Figura 3.189 Conexión de *Honeypot* 1 en *Kali Linux* en Práctica No.4

En *Kali Linux* se procede a escanear el *host* mediante la herramienta *Nmap*. El comando "*Nmap*" se acompaña de la opción "*--send-ip -P0*" para el análisis de los puertos TCP en la dirección IP 192.168.1.23, ver en la Figura 3.190.

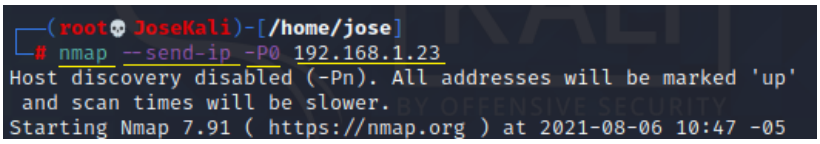


Figura 3.190 Escaneo de puertos TCP en *Honeypot* 1 en Práctica No.4

En *Honeyd* se notifica la acción del protocolo TCP como "*reset*", acción que permite que se eliminen los intentos de conexión con excepción de los puertos establecidos en el archivo de configuración, ver en la Figura 3.191.

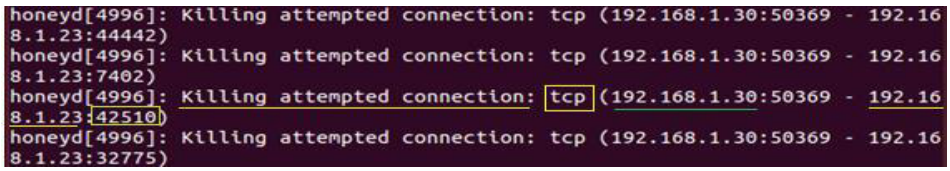


Figura 3.191 Interacción TCP en *Honeypot* 1 en Práctica No.4

El resultado del escaneo muestra a todos los puertos "abiertos" debido a la configuración de la acción del protocolo TCP en la dirección IP 192.168.1.23, ver en la Figura 3.192.

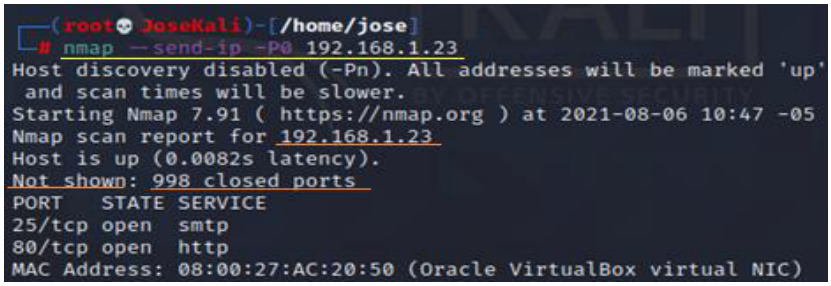


Figura 3.192 Resultados de escaneo TCP en *Honeypot* 1 en Práctica No.4

Se configuró un *script* de simulación de servidor en página web "Apache" en la dirección 192.168.1.23, se verifica en el navegador del equipo externo, ver en la Figura 3.193.

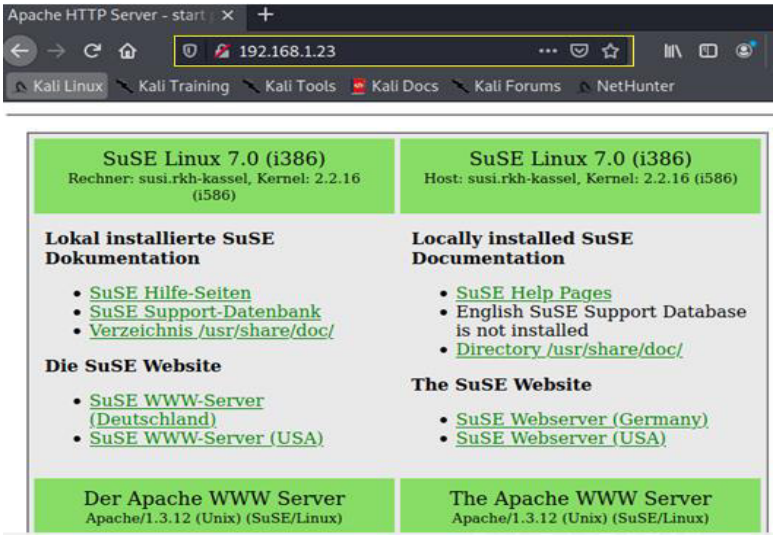


Figura 3.193 Validación de página web en Práctica No.4

Honeyd notifica el requerimiento, establecimiento y finalización de conexión por el puerto 80, ver en la Figura 3.194.

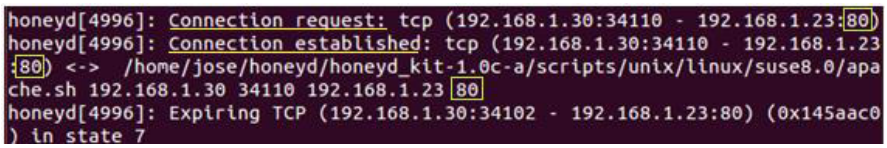


Figura 3.194 Conexión y validación de *Honeyd* con página web

Continuando, se procede con la máquina *Honeypot* "*Cisco 4500 router running IOS 11.2(2)*" mediante el comando "*ping"* en la dirección IP 192.168.1.24, ver en la Figura 3.195.

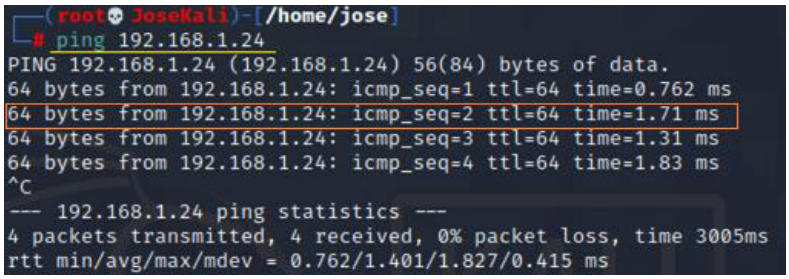


Figura 3.195 Comando "*ping*" en *Honeypot* 2 en Práctica No.4

Se realiza conexión de los mensajes ICMP, viajando desde la dirección IP "192.168.1.24" hacia el equipo externo "192.168.1.30", proceso que comprueba la creación del equipo virtual, ver en la Figura 3.196.

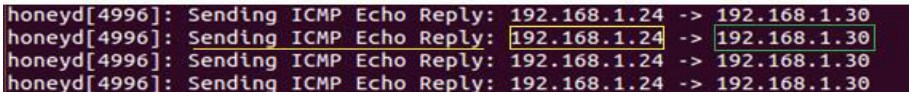


Figura 3.196 Conexión de *Honeypot* 2 en *Kali Linux* en Práctica No.4

En *Kali Linux* se procede a escanear el *host* mediante la herramienta *Nmap*. El comando "*Nmap*" se acompaña de la opción "*--send-ip -P0*" para el análisis de los puertos TCP en la dirección IP 192.168.1.24, ver en la Figura 3.197.

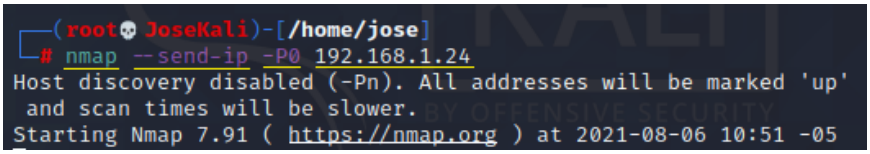


Figura 3.197 Escaneo de puertos TCP en *Honeypot* 2 en Práctica No.4

Honeyd notifica la acción del protocolo TCP como "*reset*", acción que permite que se eliminen los intentos de conexión con excepción de los puertos establecidos en el archivo de configuración, ver en la Figura 3.198.

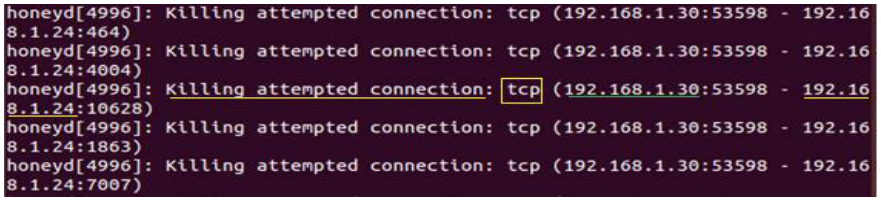


Figura 3.198 Interacción TCP de *Honeypot* 2 en Práctica No.4

El escaneo muestra puertos que se establecieron como abiertos y determinada a los demás como filtrados. El puerto 23 se encuentra abierto en servicio de Telnet, ver en la Figura 3.199.

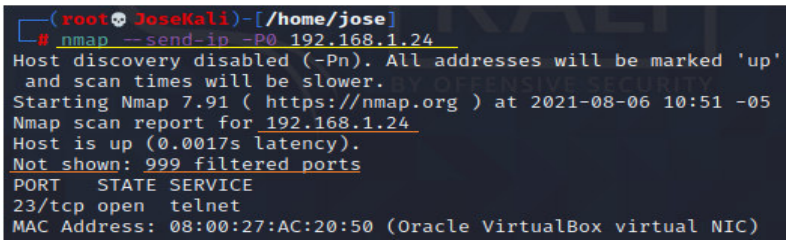


Figura 3.199 Resultados de escaneo TCP en *Honeypot* 2 en Práctica No.4

Honeyd emula la entrada al fichero del servicio con un mensaje de advertencia y comandos de ingreso, ver en la Figura 3.200.

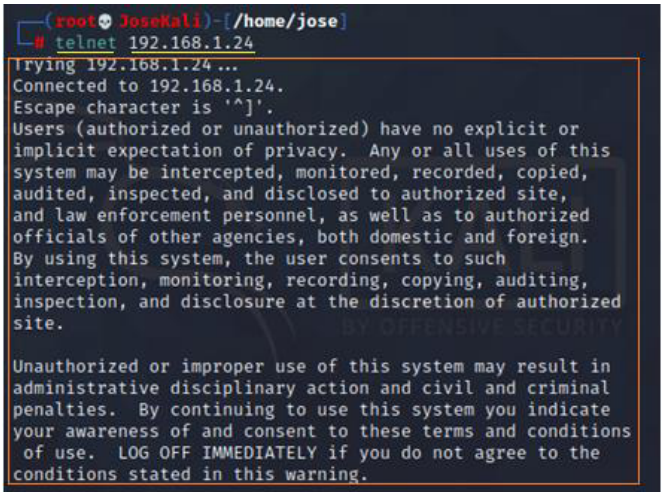


Figura 3.200 Validación de servicio Telnet

Se procede a interactuar con el entorno en base a nombres de usuario y contraseñas, ver en la Figura 3.201.

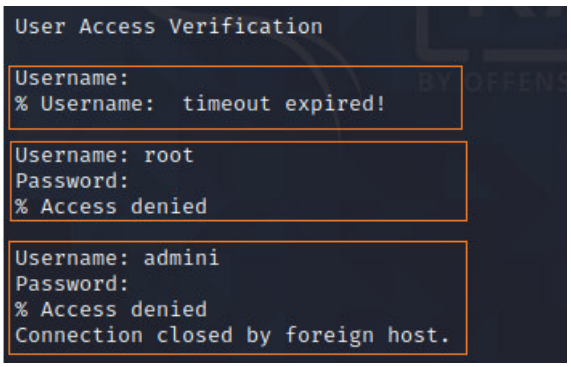


Figura 3.201 Pruebas de servicio Telnet

Nota: Cuando se procesa el servidor Telnet se debe considerar el tiempo de ejecución debido que el *Honeyd* por medio de este servicio establece un tiempo y considera como un parámetro adicional en el ingreso.

Honeyd notifica el requerimiento, establecimiento y finalización de conexión por el puerto 23. Además, notifica las credenciales empleadas en *Honeypot*, ver en la Figura 3.202.

Figura 3.202 Conexión y validación de *Honeyd* con servicio Telnet

Al final, se procede a validar la máquina *Honeypot* "*Linux Kernel 2.4.3 SMP (RedHat)"* mediante el comando "*ping*" en la dirección IP 192.168.1.25, ver en la Figura 3.203.

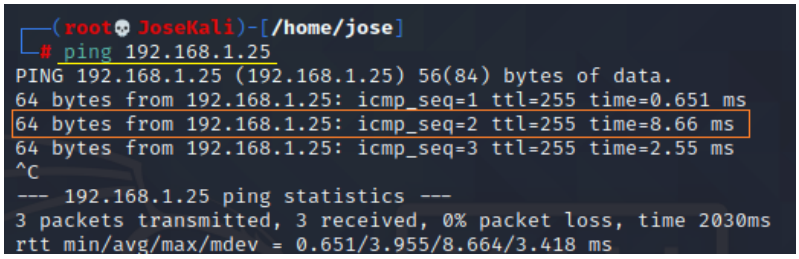


Figura 3.203 Comando "*ping*" en *Honeypot* 3 en Práctica No.4

Se realiza conexión de los mensajes ICMP, viajando desde la dirección IP 192.168.1.25 hacia el equipo externo 192.168.1.30, proceso que comprueba la creación del equipo virtual, ver en Figura 3.204.

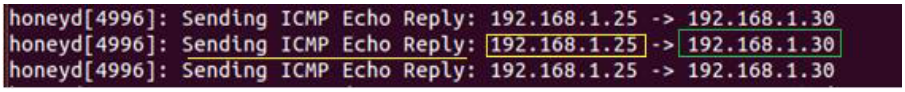


Figura 3.204 Conexión de *Honeypot* 3 en *Kali Linux* en Práctica No.4

En *Kali Linux* se procede a escanear el *host* mediante la herramienta *Nmap*. El comando "*Nmap*" se acompaña de la opción "*--send-ip -P0*" para el análisis de los puertos TCP en la dirección IP 192.168.1.25, ver en Figura 3.205

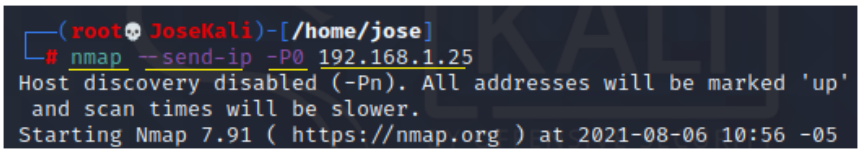


Figura 3.205 Escaneo de puertos TCP en *Honeypot* 3 en Práctica No.4

Honeyd establece la acción del protocolo TCP como "*reset*", acción que permite que se eliminen los intentos de conexión con excepción de los puertos establecidos en el archivo de configuración, ver en la Figura 3.206.

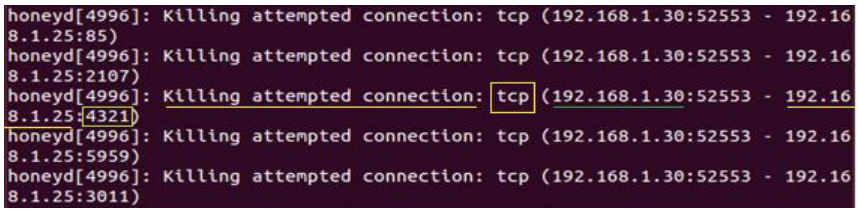


Figura 3.206 Interacción TCP de *Honeypot* 3 en Práctica No.4

El escaneo muestra puertos que se establecieron como abiertos y determinada a los demás por cerrados debido a la acción "*reset*" que se configuró previamente en el archivo, ver en la Figura 3.207.

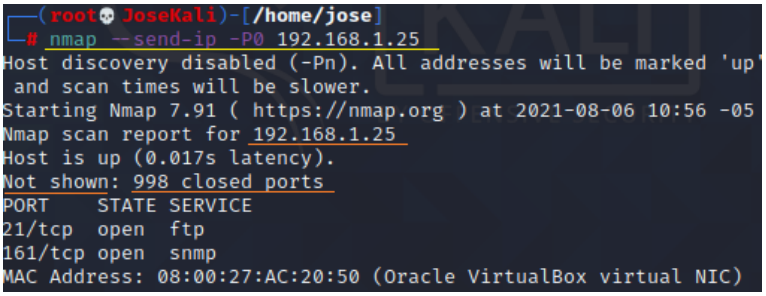


Figura 3.207 Resultados de escaneo TCP en *Honeypot* 3 en Práctica No.4

Se procede a realizar la prueba de funcionamiento de simulación del servidor FTP. El servidor virtual solicita un nombre de usuario, ver en la Figura 3.208.

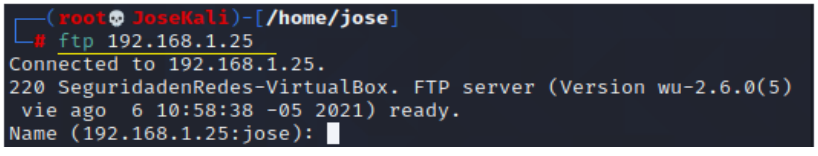


Figura 3.208 Validación de servicio FTP

Honeyd notifica el requerimiento, establecimiento y ruta del archivo *script* de simulación, ver en la Figura 3.209.

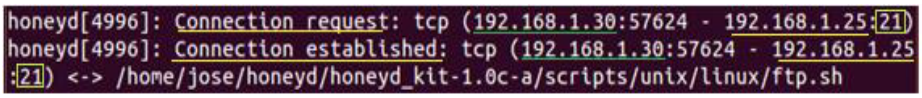


Figura 3.209 Conexión y validación de *Honeyd* con servicio FTP

Se procede a bloquear en la interfaz de la simulación y al mismo tiempo salir con el comando "*quit*". Cabe mencionar que el servicio en el cual se está simulando es un señuelo y busca ser la interfaz más cercana a la real, ver en la Figura 3.210.

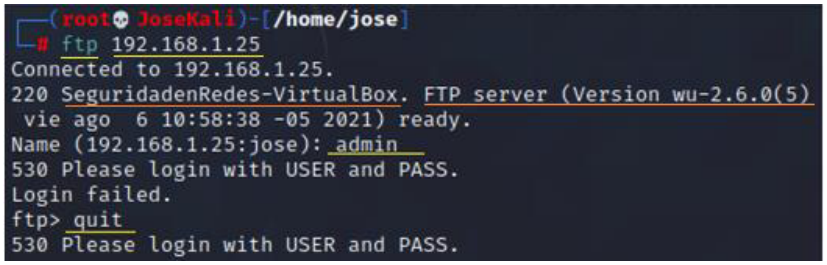


Figura 3.210 Pruebas de servicio FTP

Honeyd notifica el cierre del puerto TCP 21, ver en la Figura 3.211.

Figura 3.211 Notificación de cierre de conexión de servicio FTP

Se procede a cerrar el proceso de *Honeyd* mediante "Ctrl+ c".

Paso 9: Archivo de registro de información

Se realiza el ingreso del archivo de registro mediante el comando "*vim"* acompañado de la ruta *"/var/log/honeyd/"* "con el nombre del archivo "*honeyd4.log*", ver en la Figura 3.212.

Figura 3.212 Archivo de registro en Práctica No.4

El archivo muestra el análisis y los datos proporcionados por *Honeyd* a la respuesta de la Red *Honeypot*. A continuación, se detallan los datos registrados en cada una de las personalidades:

- **Análisis *Honeypot* "*Linux 2.4.7 (X86)".***
  El análisis de mensajes ICMP y puertos TCP se visualiza en las Figura 3.213 y Figura 3.214.

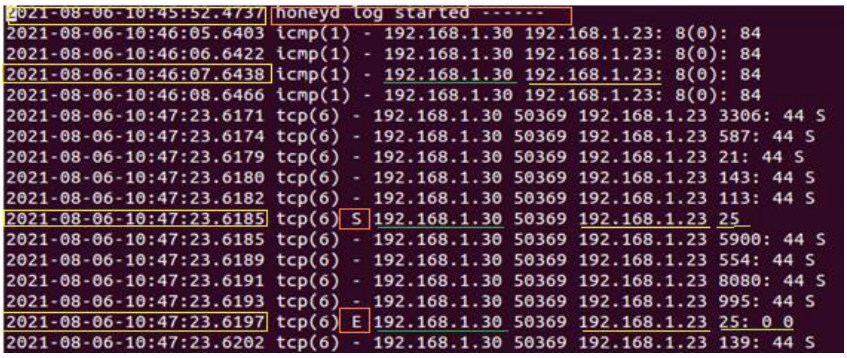


Figura 3.213 Análisis TCP de *Honeypot* 1 en Práctica No.4

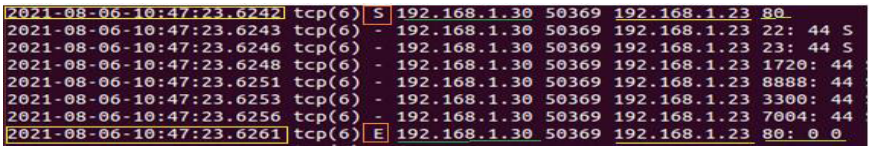


Figura 3.214 Análisis 2 TCP de *Honeypot* 1 en Práctica No.4

1. Fecha y hora de Inicialización del proceso de *Honeyd.*
2. Mensaje “ICMP” enviado entre *host*.
3. Análisis de puertos TCP.
4. Inicio de conexión (S) en grupo de puertos TCP.
5. Finalización de conexión (E) en grupo de puertos TCP.

Análisis de comportamiento en página Web, ver en la Figura 3.215.

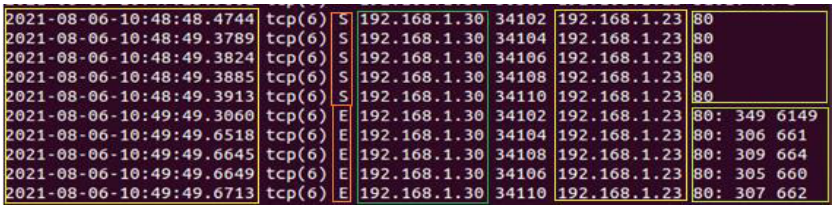


Figura 3.215 Análisis TCP de servicio de página Web

1. Establecimiento de conexión (S) de puerto TCP 80.
2. Finalización de conexión (E) de puerto TCP 80.
3. Valor de paquetes enviados de fuente a destino.

- **Análisis *Honeypot* “*Cisco 4500 router running IOS 11.2(2)*”**

  El análisis de mensajes ICMP y puertos TCP se visualiza en las Figura 3.216 y Figura 3.217.

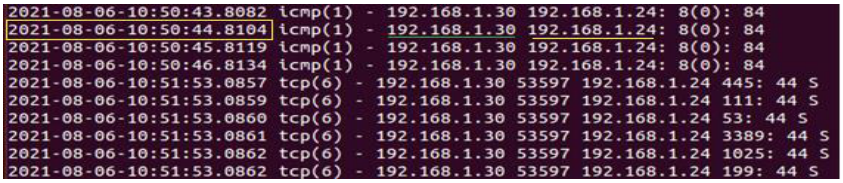


Figura 3.216 Análisis TCP de *Honeypot* 2 en Práctica No.4

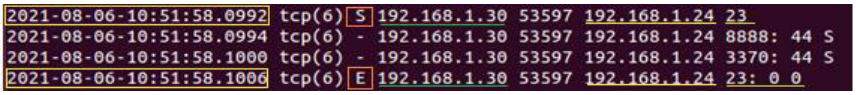


Figura 3.217 Análisis 2 TCP de *Honeypot* 2 en Práctica No.4

1. Fecha y hora de proceso de *Honeyd.*
2. Mensaje “ICMP” enviado entre *host*.
3. Análisis de puerto TCP.
4. Inicio de conexión (S) y finalización de conexión (E) en puerto TCP 23.

Análisis de comportamiento en servidor Telnet, ver en la Figura 3.218.

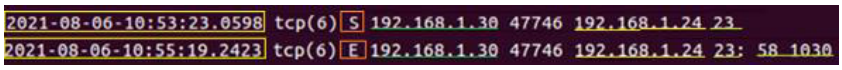


Figura 3.218 Análisis de servicio Telnet

1. Inicio de conexión (S) de puerto TCP 23.
2. Finalización de conexión (E) de puerto TCP 23.
3. Valor de paquetes enviados de fuente a destino.

- **Análisis *Honeypot* “*Linux Kernel 2.4.3 SMP (RedHat)”***

  El análisis de mensajes ICMP y puertos TCP se visualiza en la Figura 3.219.

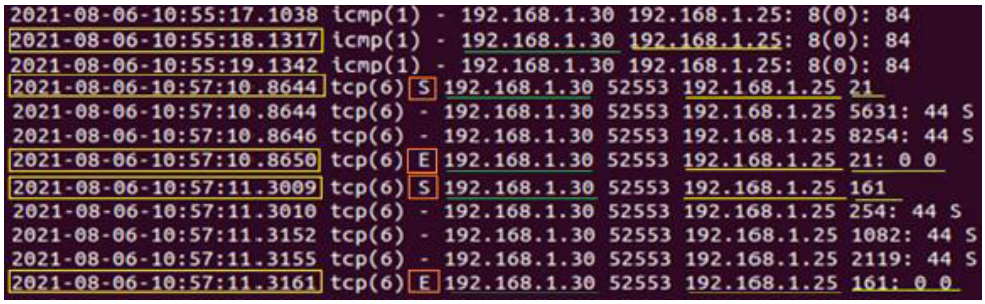


Figura 3.219 Análisis TCP de *Honeypot* 3 en Práctica No.4

1. Fecha y hora de proceso de *Honeyd*.
2. Sin mensaje “ICMP”.
3. Análisis de puertos TCP por *Nmap*.

4. Inicio de conexión (S) y finalización de conexión (E) en grupo de puertos TCP 21 y 161.

Análisis de comportamiento en servidor FTP, ver en la Figura 3.220.

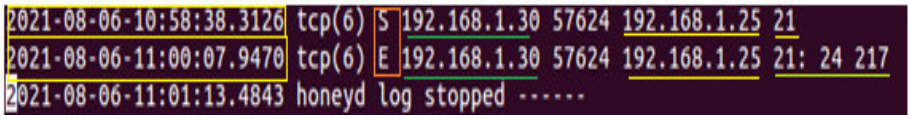


Figura 3.220 Análisis de servicio FTP

1. Inicio de conexión (S) y finalización de conexión (E) en puerto TCP 21.
2. Valor de paquetes enviados de fuente a destino.

3. CONCLUSIONES

- El conjunto de personalidades y servicios se ajusta al establecimiento de señuelos sobre la red; es decir, permite absorción de información del atacante sin poner en riesgo un equipo real.
- El archivo de configuración "*honeyd4.conf*" mediante la asignación del nombre "s*use80*" permite la creación del *Honeypot* en la personalidad "*Linux 2.4.7 (X86)",* el encargado de simular el estado de una página web en la Red *Honeypot.*
- La personalidad "*Cisco 4500 router running IOS 11.2(2*)" se encarga de la simulación de un Servidor TELNET en la red, permitiendo ajustar el acceso por medio de credenciales y tiempo de espera sobre el *Honeypot.*

4. RECOMENDACIONES

- Se recomienda ajustar el envío de mensajes ICMP en el equipo *Kali Linux*; con la capacidad mínima de 4 mensajes por personalidad, con el fin de regular al sistema y permitir que el proceso sea más ágil sin interrupciones.
- Se sugiere tomar el tiempo necesario para la simulación de la página web, servidor TELNET y servidor FTP. Debido a que *Honeyd* necesita obtener la información del *Honeypot* referente al establecimiento de conexión, puertos y tiempo de ejecución.
- En el archivo de registro "*honeyd4.log*" se recomienda validar la respuesta del escaneo por la herramienta *Nmap* y la simulación de los servicios en los

puertos TCP 80, 23 y 161. La herramienta *Nmap* establece la conexión sin paquetes en comparación de la simulación de servicio.

PRÁCTICA 1

1. **TEMA:** Introducción a los *Honeypots*
2. **OBJETIVOS**
   - **2.1. OBJETIVO GENERAL**
     - Familiarizar al estudiante sobre el uso de los Sistemas de Seguridad "*Honeypots*" y ejecutar una serie de comandos para crear un *Honeypot* simple.
   - **2.2. OBJETIVOS ESPECÍFICOS**
     - Interactuar con el Sistema Operativo *GNU/Linux Ubuntu*.
     - Conocer la herramienta *Honeyd* y sus principales componentes.
     - Ejecutar comandos para la configuración de un *Honeypot* simple.
     - Visualizar los resultados obtenidos.
3. **TRABAJO PREPARATORIO**
   - **3.1. CUESTIONARIO**
     - Realizar un resumen sobre los sistemas de Seguridad *Honeypots*.
     - Realizar un resumen sobre la herramienta *Honeyd*.
     - Enumerar las características más importantes de los siguientes componentes:
       - *arpd*
       - *Nmap.prints*
       - *Nmap.assoc*
       - *xprobe2.conf*
       - *pf.os*
4. **DESCRIPCIÓN DE ACTIVIDADES Y PROCEDIMIENTO DE LA PRÁCTICA**

   4.1. Encender el programa *Oracle VirtualBox* y seleccionar la máquina virtual *Honeyd_Ubuntu*.

   4.2. Ingresar al sistema con la contraseña "jose".

   4.3. Seleccionar el icono de la terminal e ingresar al usuario "root" mediante el comando "***sudo su***" y la contraseña del sistema "***jose***".

   4.4. Ingresar al directorio de *Honeyd* en la ruta *"/home/jose/honeyd/honeyd_kit-1.0c-a*" mediante el comando "*cd*".

   cd /home/jose/honeyd/honeyd_kit-1.0c-a

4.5. Crear una copia del archivo de configuración dentro del directorio "*honeyd*" con el nombre "*honeyd1.conf*" mediante el comando "*cp*".

cp honeyd.conf /home/jose/honeyd/honeyd_kit-1.0c-a/honeyd1.conf

4.6. Visualizar y analizar la topología de Red *Honeypot*, ver en la Figura 3.221.

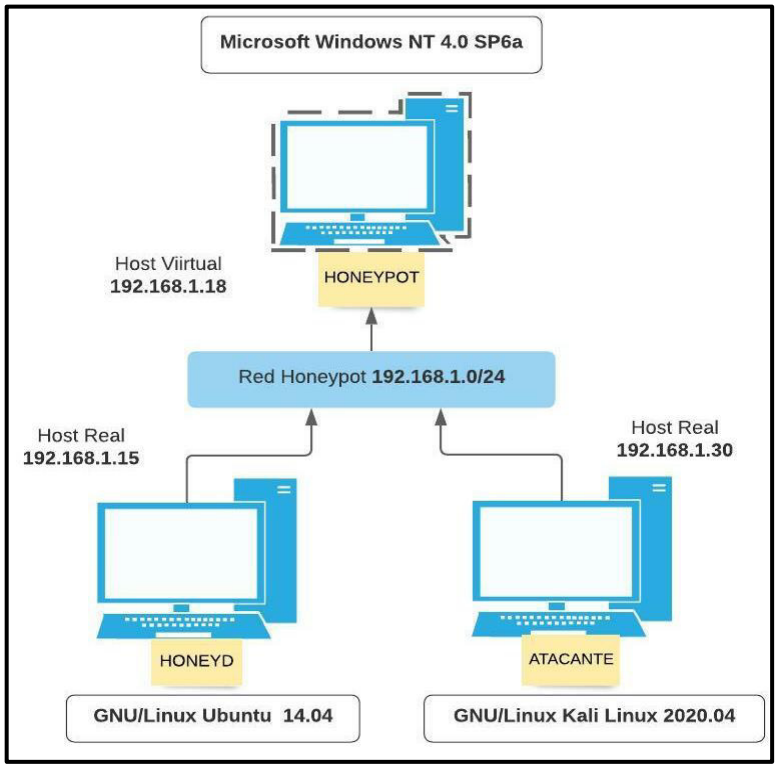


Figura 3.221 Topología de red No.1

4.7. Verificar la dirección IP del equipo mediante el comando "*ifconfig*".

4.8. Si no se encuentra se procede a asignar la dirección IP del equipo a la dirección 192.168.1.15 por medio del comando "*ifconfig*".

ifconfig eth0 192.168.1.15 netmask 255.255.255.0

4.9. Elegir el sistema operativo o personalidad en el archivo "*Nmap.assoc*" mediante el comando "*vim*".

vim nmap.assoc

4.10. Seleccionar el sistema operativo "***Microsoft Windows NT 4.0 SP6a***" mediante la visualización.

:/Microsoft Windows NT 4.0 SP6a

4.11. Salir del archivo mediante la tecla "***Esc***" y escribir a opción "***:wq***".

4.12. Ingresar al archivo de configuración "*honeyd1.conf*" mediante el comando "*vim*"

vim honeyd1.conf

4.13. Argumentos del archivo de configuración de "*honeyd*", ver en la Tabla 3.15.

Tabla 3.15 Argumentos de configuración de Práctica No.1 [27] [32]

Argumento	Función
create	Creación de Personalidad o Sistema Operativo
set	Establecer acción
default	Por defecto
add	Agregar función o puerto
TCP/UDP	Protocolos
port	Puerto
open	Abierto
reset	*Reseteo*
block	Bloqueo
bind	Enlace o Dirección IP

4.14. Presionar la letra **"i"** en el teclado para configurar y editar el archivo de configuración.

4.15. Implementar la configuración en el apartado final del archivo de configuración "***honeyd1.conf***", ver en la Figura 3.222.

```
### Creación de Honeypot Simple "Windows"

create windows
set windows personality "Microsoft Windows NT 4.0 SP6a"
set windows default tcp action reset
set windows default udp action reset
set windows default icmp action open

bind 192.168.1.18 windows
```

Figura 3.222 Configuración de archivo en Práctica No.1

4.16. Guardar el archivo de configuración mediante la tecla "***Esc***" y escribir a opción "***:wq***".

4.17. Arranque *honeyd.* Se inicia con el arranque del demonio "*arpd*" mediante el *script* "*start-arpd.sh*" con el comando "*./*".

./start-arpd.sh

4.18. Arranque del demonio de "*Honeyd*" mediante el *script* "*star-Honeyd.sh*" con el comando "*./*".

./star-honeyd.sh

4.19. Argumentos de arranque de Honeyd, ver en la Tabla 3.16.

Tabla 3.16 Comandos de arranque de Práctica No.1 [27] [32]

Opción	Función
Honeyd	Arranque de *Honeyd*
-d	Levantamiento del Demonio
-i	Tarjeta de Red "eth0"
-f	Archivo de configuración- ruta "*Honeyd1.conf*"
-p	Archivo "*Nmap.prints*" conjunto de personalidades-ruta
-x	Archivo "*xprobe2.conf*" sonda-lector de personalidades-ruta
-a	Archivo "*Nmap.assoc*" personalidades-sonda-ruta
-0	Archivo "*pf.os*" lector de personalidades pasivas-ruta
-l	Archivo de Registro de Información "*Honeyd1.log*"
192.168.1.18	Dirección IP del *Honeypot*

4.20. Escribir las opciones del comando "honeyd", ver en la Figura 3.223.

```
# honeyd -d -i eth0 -f honeyd1.conf -p nmap.prints -x xprobe2.conf -a
nmap.assoc -0 pf.os -l /var/log/honeyd/honeyd1.log 192.168.1.18
```

Figura 3.223 Comandos de ejecución de Práctica No.1

4.21. Se debe dirigir al programa *Oracle VirtualBox* y seleccionar la máquina virtual *Honeyd_ Kali Linux*.

4.22. Se debe ingresar lo antes posible al equipo *Kali Linux* e ingresar las credenciales del Usuario "***jose***" y Contraseña "***josemegadeth***"

4.23. Ingresar al icono de la terminal en la esquina superior izquierda. Además, se debe ingresar al usuario "*root*" mediante el comando "***sudo su***" y la misma contraseña.

4.24. Se debe dirigir al icono cuadrado de red ubicado en la esquina superior derecha, junto al icono de reloj en el escritorio de *Kali Linux* y validar que las tarjetas de red se encuentren en el orden *"Conexión cableada 1*" y "*Conexión cableada 2*", de no ser así el caso, corregir el error.

4.25. Comprobar la valides de la tarjeta de red en "eth0" con la dirección IP 192.168.1.30.

4.26. Regresar a la maquina *Honeyd_ Ubuntu* y correr el comando "honeyd" previamente escrito presionando en el teclado "**Enter**".

4.27. Se debe verificar la creación del *Honeypot* en el equipo *Kali Linux* mediante el comando "*ping*" en la dirección IP.

ping 192.168.1.18

4.28. Visualizar la respuesta de *Honeyd* en ejecución, ver la Figura 3.224.

Figura 3.224 Conexión de Práctica No.1

4.29. Se procede a detener el comando "*ping*" en la máquina "atacante" mediante la opción "***Ctrl + c***" en el teclado.

4.30. Se procede a detener el demonio de *Honeyd* mediante "***Ctrl + c***" en el teclado.

4.31. Visualizar el archivo de registro en el archivo "*honeyd1.log*" en el directorio "*/var/log/honeyd/*" mediante el comando "*vim*" y una vez finalizado salir con la opción "Esc + :q".

vim /var/log/honeyd/honeyd1.log

4.32. Visualizar los resultados en el archivo de registro en la Tabla 3.17.

Tabla 3.17 Resultado de Práctica No.1

Hora Evento	Protocolo	Conexión	Dir. IP Fuente-Destino		Tamaño Paquete
2021-07-29-05:07:38.9637	*Honeyd* log started -------				
2021-07-29 05:08:10.1553	ICMP (1)	-	192.168.1.30 192.168.1.18	8(0)	84
…….					
2021-07-29-05:09:34.2051	*Honeyd* log stopped -------				

5. INFORME

5.1. Investigar los métodos de seguridad o encubrimiento de direcciones IP.

5.2. Presentar el análisis de los resultados obtenidos del archivo de registro de información "*honeyd1.log*".

ESCUELA POLITÉCNICA NACIONAL
ESCUELA DE FORMACIÓN DE TECNÓLOGOS
HOJA GUÍA PARA ESTUDIANTE

PRÁCTICA 2

1. **TEMA:** Validación de Puertos TCP y UDP.
2. **OBJETIVOS**

2.1. OBJETIVO GENERAL

- Aprender a configurar un Sistema de Seguridad "*Honeypot*" ejecutando comandos para evaluar los puertos de los protocolos (TCP y UDP).

2.2. OBJETIVOS ESPECÍFICOS

- Interactuar con el Sistema Operativo *GNU/Linux Ubuntu*.
- Manejar la herramienta *Honeyd* y sus principales componentes.
- Ejecutar comandos para la configuración de un *Honeypot* en base a puertos TCP y UDP.
- Visualizar los resultados obtenidos.

3. **TRABAJO PREPARATORIO**

3.1. CUESTIONARIO

- ¿Qué es el protocolo TCP?
- ¿Qué es el protocolo UDP?
- Definir la pila o el modelo TCP/IP.
- ¿Qué es un puerto?
- Definir la herramienta *Nmap.*
- Investigar los siguientes puertos:
  - TCP: 135,139 y 445.
  - UDP: 135,137 y 4500.

4. **DESCRIPCIÓN DE ACTIVIDADES Y PROCEDIMIENTO DE LA PRÁCTICA**

4.1. Considerar las condiciones de ingreso a la máquina virtual de *Honeyd_ Ubuntu* de la Práctica No.1.

4.2. Crear una copia del archivo de configuración dentro del directorio "*honeyd*" con el nombre "*honeyd2.conf*" mediante el comando "*cp*".

cp honeyd.conf /home/jose/honeyd/honeyd_kit-1.0c-a/honeyd2.conf

4.3. Visualizar y analizar la topología de Red *Honeypot*, ver en la Figura 3.225.

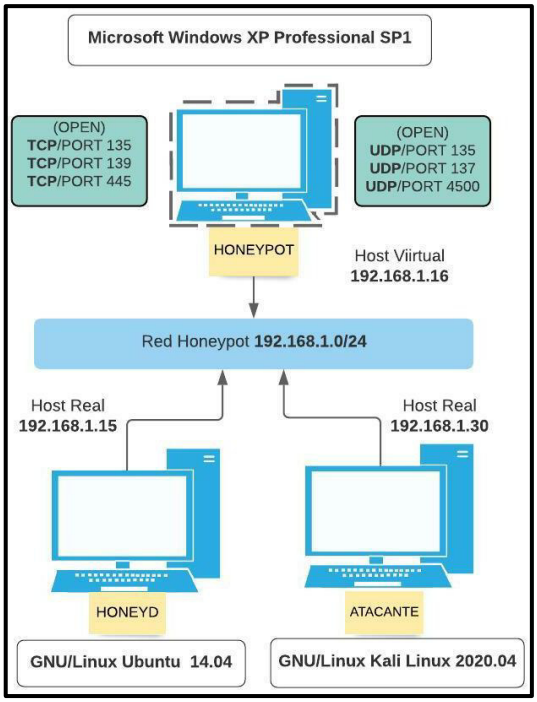


Figura 3.225 Topología de red 2

4.4. Selección de personalidad en archivo "*Nmap.assoc*" mediante el comando "*vim"*

vim nmap.assoc

4.5. Seleccionar el sistema operativo "***Microsoft Windows XP Professional SP1*** "

4.6. Ingresar al archivo de configuración "*honeyd2.conf*" mediante el comando "*vim*".

vim honeyd2.conf

4.7. Argumentos del archivo de configuración de "*Honeyd*" en la Tabla 3.18.

Tabla 3.18 Líneas de comandos de configuración de Práctica No.2 [27] [32]

Argumento	Función
create	Creación de Personalidad o Sistema Operativo
set	Establecer acción
default	Por defecto
add	Agregar función o puerto
TCP/UDP	Protocolos
port	Puerto
open	Abierto
reset	*Reseteo*
block	Bloqueo

bind	Enlace o dirección IP

4.8. Implementar configuración en el apartado final del archivo de configuración "***honeyd2.conf***", ver en la Figura 3.226.

```
### Creación de Plantilla de "Honeypot" y puertos

create windows
set windows personality "Microsoft Windows XP Professional SP1"
set windows default tcp action reset
set windows default udp action reset
set windows default icmp action open

add windows tcp port 135 open
add windows tcp port 139 open
add windows tcp port 445 open
add windows udp port 135 open
add windows udp port 137 open
add windows udp port 4500 open

bind 192.168.1.16 windows
```

Figura 3.226 Configuración de Práctica No.2

4.9. Guardar el archivo mediante la tecla "***Esc***" y escribir la opción "***:wq***".

4.10. Arranque *arpd.* Se inicia con el arranque del demonio "*arpd*" mediante el *script* "*start-arpd.sh*" con "*./*".

./start-arpd.sh

4.11. Arranque del demonio de "*honeyd*" mediante el *script* "*star-Honeyd.sh*" con el comando "*./*".

./star-honeyd.sh

4.12. Argumentos de arranque de *Honeyd* en la Tabla 3.19.

Tabla 3.19 Argumentos de configuración de Práctica No.2 [27] [32]

Opción	Función
honeyd	Arranque de *Honeyd*
-d	Levantamiento del Demonio
-i	Tarjeta de Red "eth0"
-f	Archivo de configuración- ruta "*Honeyd2.conf*"
-p	Archivo "*Nmap.prints*" conjunto de personalidades-ruta
-x	Archivo "*xprobe2.conf*" sonda-lector de personalidades-ruta
-a	Archivo "*Nmap.assoc"* personalidades-sonda-ruta
-0	Archivo "*pf.os*" lector de personalidades pasivas-ruta

-l	Archivo de Registro de Información "*Honeyd2.log*"
192.168.1.16	Dirección IP del *Honeypot*

4.13. Escribir las opciones del comando "honeyd", ver en la Figura 3.227.

```
# honeyd -d -i eth0 -f honeyd2.conf -p nmap.prints -x xprobe2.conf -a
nmap.assoc -0 pf.os -l /var/log/honeyd/honeyd2.log 192.168.1.16
```

Figura 3.227 Comando de ejecución de Práctica No.2

4.14. Validación de demonio. Se debe considerar las mismas condiciones del ingreso y procesamiento del equipo *Kali Linux* de la Práctica No.1.

4.15. Se debe verificar la creación del *Honeypot* mediante el comando "*ping*".

ping 192.168.1.16

4.16. Visualizar la respuesta de *Honeyd* en ejecución, ver en la Figura 3.228.

```
honeyd ["#proceso"]: Sending ICMP Echo Reply: 192.168.1.16 ->192.168.1.30
```

Figura 3.228 Conexión de Práctica No.2

4.17. Se procede a detener el comando "*ping*" en la máquina "atacante" mediante la opción "***Ctrl + c***" en el teclado.

4.18. Se procede a escanear los puertos TCP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-P0*".

nmap –send-ip -P0 192.168.1.16

4.19. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.229.

```
honeyd ["#proceso"]: Killing attempted connection: tcp (192.168.1.30:
#puerto origen - 192.168.1.16: #puerto destino)
```

Figura 3.229 Interacción TCP en Práctica No.2

4.20. Se procede a escanear los puertos UDP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-sU*".

nmap –send-ip -sU 192.168.1.16

4.21. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.230.

honeyd ["#proceso"]: **Connection to closed port**: udp (192.168.1.30: #puerto origen - 192.168.1.16: #puerto destino)

Figura 3.230 Interacción UDP en Práctica No.2

4.22. Se procede a detener el demonio de *Honeyd* mediante "***Ctrl + c***" en el teclado.

4.23. Visualizar el archivo de registro en el archivo "*honeyd2.log*" en el directorio "*/var/log/honeyd/"* mediante el comando "*vim*"

vim /var/log/honeyd/honeyd2.log

4.24. Visualizar los resultados del archivo de registro .

Los resultados de mensaje de conexión se visualizan en la Figura 3.231.

```
2021-08-10-21:36:48.3415 honeyd log started ------
2021-08-10-21:36:49.1499 icmp(1) - 192.168.1.30 192.168.1.16: 8(0): 84
2021-08-10-21:36:50.1557 icmp(1) - 192.168.1.30 192.168.1.16: 8(0): 84
2021-08-10-21:36:51.1613 icmp(1) - 192.168.1.30 192.168.1.16: 8(0): 84
2021-08-10-21:36:52.1676 icmp(1) - 192.168.1.30 192.168.1.16: 8(0): 84
2021-08-10-21:36:53.1722 icmp(1) - 192.168.1.30 192.168.1.16: 8(0): 84
```

Figura 3.231 Resultado ICMP en Práctica No.2

1. Fecha y hora de Inicialización del proceso de *Honeyd.*
2. Mensaje "ICMP" enviado entre *host.*

Los resultados de puertos TCP se visualizan en la Figura 3.232.

```
2021-08-10-21:38:03.5624 tcp(6) S 192.168.1.30 60571 192.168.1.16 445
2021-08-10-21:38:03.5626 tcp(6) - 192.168.1.30 60571 192.168.1.16 8888: 44 S
2021-08-10-21:38:03.5628 tcp(6) - 192.168.1.30 60571 192.168.1.16 23: 44 S
2021-08-10-21:38:03.5631 tcp(6) - 192.168.1.30 60571 192.168.1.16 1025: 44 S
2021-08-10-21:38:03.5633 tcp(6) - 192.168.1.30 60571 192.168.1.16 22: 44 S
2021-08-10-21:38:03.5639 tcp(6) S 192.168.1.30 60571 192.168.1.16 135
2021-08-10-21:38:03.5640 tcp(6) - 192.168.1.30 60571 192.168.1.16 1723: 44 S
2021-08-10-21:38:03.5643 tcp(6) E 192.168.1.30 60571 192.168.1.16 445: 0 0
2021-08-10-21:38:03.5644 tcp(6) - 192.168.1.30 60571 192.168.1.16 5900: 44 S
2021-08-10-21:38:03.5647 tcp(6) - 192.168.1.30 60571 192.168.1.16 25: 44 S
2021-08-10-21:38:03.5658 tcp(6) E 192.168.1.30 60571 192.168.1.16 135: 0 0
2021-08-10-21:38:03.5661 tcp(6) S 192.168.1.30 60571 192.168.1.16 139
2021-08-10-21:38:03.5663 tcp(6) - 192.168.1.30 60571 192.168.1.16 993: 44 S
2021-08-10-21:38:03.5666 tcp(6) - 192.168.1.30 60571 192.168.1.16 554: 44 S
2021-08-10-21:38:03.5670 tcp(6) - 192.168.1.30 60571 192.168.1.16 199: 44 S
2021-08-10-21:38:03.5673 tcp(6) - 192.168.1.30 60571 192.168.1.16 21: 44 S
2021-08-10-21:38:03.5677 tcp(6) - 192.168.1.30 60571 192.168.1.16 8080: 44 S
2021-08-10-21:38:03.5681 tcp(6) - 192.168.1.30 60571 192.168.1.16 995: 44 S
2021-08-10-21:38:03.5685 tcp(6) - 192.168.1.30 60571 192.168.1.16 1151: 44 S
2021-08-10-21:38:03.5692 tcp(6) - 192.168.1.30 60571 192.168.1.16 9000: 44 S
2021-08-10-21:38:03.5702 tcp(6) E 192.168.1.30 60571 192.168.1.16 139: 0 0
2021-08-10-21:38:03.5737 tcp(6) - 192.168.1.30 60571 192.168.1.16 5298: 44 S
```

Figura 3.232 Resultado TCP en Práctica No.2

1. Análisis de puertos TCP.

2. Inicio de conexión (S) en puertos TCP.
3. Finalización de conexión (E) en puertos TCP.

Los resultados de puertos UDP se visualizan en la Figura 3.233.

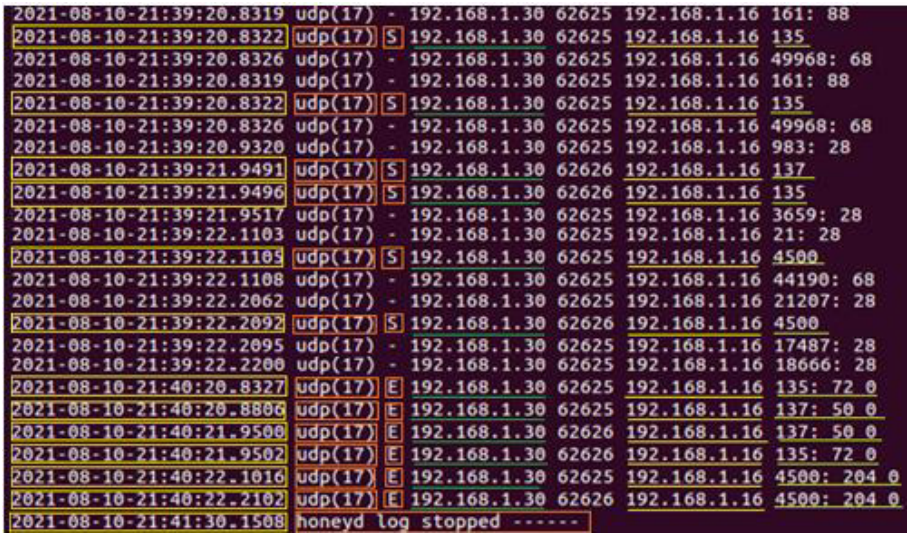


Figura 3.233 Resultado UDP en Práctica No.2

1. Análisis de puertos UDP.
2. Inicio de conexión (S) en puertos UDP.
3. Finalización de conexión (E) en puertos UDP.
4. Finalización del proceso de *Honeyd*.

5. INFORME

5.1. Investigar los métodos más comunes de infiltración por medio de puertos.

5.2. Presentar el análisis de los resultados obtenidos del archivo de registro de información "*honeyd2.log*".

PRÁCTICA 3

1. **TEMA:** Acciones de Protocolos TCP, UDP e ICMP
2. **OBJETIVOS**

2.1. OBJETIVO GENERAL

- Identificar los principales mecanismos de protección de los Sistemas de Seguridad "*Honeypots*" y evaluar su procesamiento en base a la configuración de comandos.

2.2. OBJETIVOS ESPECÍFICOS

- Interactuar con Sistemas Operativos *GNU/Linux Ubuntu*.
- Ejecutar una serie de comandos para obtención de un conjunto de *Honeypots*.
- Configurar acciones de Protocolo TCP, UDP e ICMP.
- Visualizar los resultados obtenidos.

3. **TRABAJO PREPARATORIO**

3.1. CUESTIONARIO

- Enumerar las características de los mensajes *SYN/ACK*
- Realizar un resumen sobre los mensajes *RST*
- Investigar los siguientes puertos:
  - TCP: 80, 22, 23, 25, 21 y 2222.
  - UDP: 135,137 y 53.

4. **DESCRIPCIÓN DE ACTIVIDADS Y PROCEDIMIENTO DE LA PRÁCTICA**

4.1. Considerar las mismas condiciones de ingreso a la máquina virtual de *Honeyd_ Ubuntu* de la Práctica No.1.

4.2. Crear una copia del archivo de configuración dentro del directorio "*honeyd*" con el nombre "*honeyd3.conf*" mediante el comando "*cp*".

cp honeyd.conf /home/jose/honeyd/honeyd_kit-1.0c-a/honeyd3.conf

4.3. Visualizar y analizar la topología de Red *Honeypot* en la Figura 3.234.

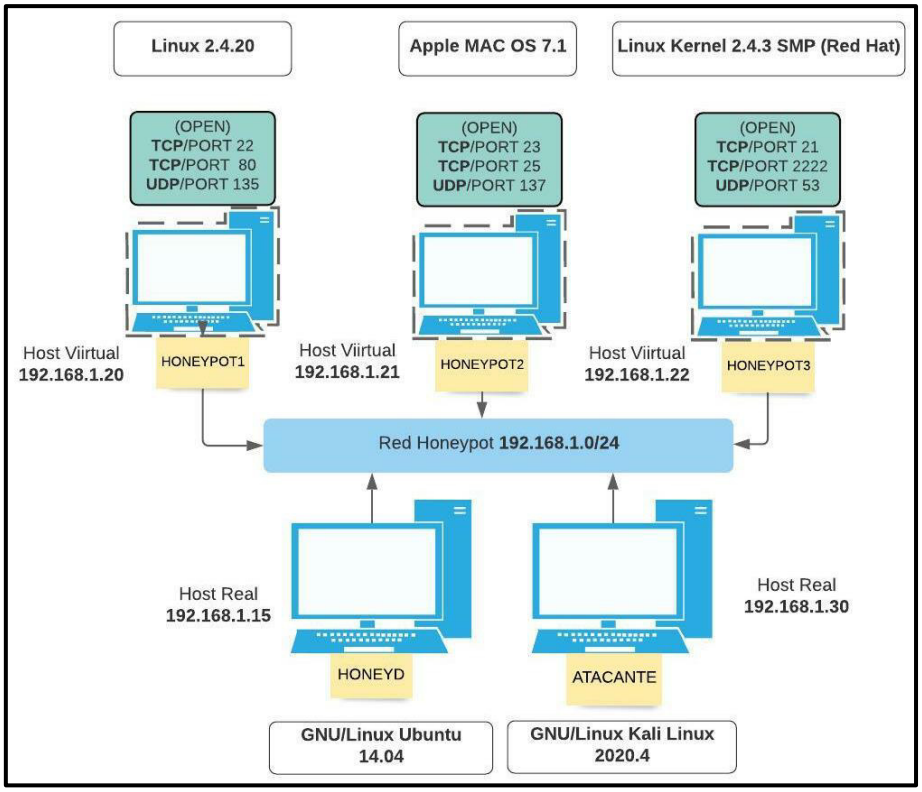


Figura 3.234 Topología de red 3

4.4. Elegir el sistema operativo o personalidad en el archivo "*Nmap.assoc*" mediante el comando "*vim*"

vim nmap.assoc

4.5. Seleccionar los sistemas operativos "***Linux 2.4.20, Apple MAC OS 7.1 y Linux Kernel 2.4.3 SMP (RedHat)*** "

4.6. Se procede a establecer las direcciones IP de las personalidades, ver en la Tabla 3.20.

Tabla 3.20 Direcciones IP de Práctica No.3.

Personalidad	Dirección IP
Linux 2.4.20	192.168.1.20
Apple MAC OS 7.1	192.168.1.21
Linux Kernel 2.4.3 SMP (Red Hat)	192.168.1.22

4.7. Ingresar al archivo de configuración "*honeyd3.conf*" mediante el comando "vim".

vim honeyd3.conf

4.8. Argumentos de configuración de *Honeyd* en la Tabla 3.21.

Tabla 3.21 Argumentos de configuración de Práctica No.3 [27] [32]

Argumento	Función
create	Creación de Personalidad o Sistema Operativo
set	Establecer acción
default	Por defecto
add	Agregar función o puerto
TCP/UDP	Protocolos
port	Puerto
open	Abierto
reset	*Reseteo*
block	Bloqueo
bind	Enlace o Dirección IP

4.9. Se procede a visualizar el despliegue de acciones en la Tabla 3.22.

Tabla 3.22 Operaciones de protocolos en archivo configuración de red 3 [27] [32]

TCP	
Acción	**Respuesta**
Open	Responde con *Syn/Ack* y establece la conexión (Abierto)
Block	Descarta el paquete y no responde
Reset	Responde con *RST* (Elimina cualquier intento de conexión)
UDP	
Acción	**Respuesta**
Open	No responde (Cerrado)
Block	Descarta el paquete y no responde
Reset	Responde con mensaje ICMP de error de puerto
Puerto ICMP	
Acción	**Respuesta**
Open	Responde a los paquetes ICMP (Abierto)
Block	Descarta el paquete y no responde

4.10. Implementar la configuración en el apartado final del archivo de configuración "***honeyd3.conf***", ver en la Figura 3.235.

```
### Plantilla de Honeypot "Linux"
create linux
set linux personality "Linux 2.4.20"
set linux default tcp action open
set linux default udp action open
set linux default icmp action open
add linux tcp port 80 open
add linux tcp port 22 open
add linux udp port 135 open

bind 192.168.1.20 linux

### Plantilla de Honeypot "Apple"
create apple
set apple personality "Apple Mac OS 7.1"
set apple default tcp action block
set apple default udp action block
set apple default icmp action open
add apple tcp port 23 open
add apple tcp port 25 open
add apple udp port 137 open

bind 192.168.1.21 apple

### Plantilla de Honeypot "Kernel"
create kernet
set kernet personality "Linux Kernel 2.4.3 SMP (RedHat)"
set kernet default tcp action reset
set kernet default udp action reset
set kernet default icmp action block
add kernet tcp port 21 open
add kernet tcp port 2222 open
add kernet udp port 53 open

bind 192.168.1.21 kernet
```

Figura 3.235 Configuración en Práctica No.3

4.11. Guardar el archivo de configuración mediante la tecla "***Esc***" y escribir a opción "***:wq***".

4.12. Arranque *Honeyd.* Se inicia con el arranque del demonio "*arpd*" mediante el *script* "*start-arpd.sh*" con el comando "*./*".

./start-arpd.sh

4.13. Arranque del demonio de "*Honeyd*" mediante el *script* "*star-Honeyd.sh*" con el comando "*./*".

./star-honeyd.sh

4.14. Argumentos de arranque de *Honeyd* en la Tabla 3.23.

Tabla 3.23 Comandos de ejecución de Práctica No.3 [27] [32]

Opción	Función
honeyd	Arranque de *Honeyd*
-d	Levantamiento del Demonio
-i	Tarjeta de Red "eth0"
-f	Archivo de configuración- ruta "*Honeyd3.conf*"
-p	Archivo "*Nmap.prints*" conjunto de personalidades-ruta
-x	Archivo "*xprobe2.conf*" sonda-lector de personalidades-ruta
-a	Archivo "*Nmap.assoc*" personalidades-sonda-ruta
-0	Archivo "*pf.os*" lector de personalidades pasivas-ruta
-l	Archivo de Registro de Información "*Honeyd3.log*"
192.168.1.20-192.168.1.22	Dirección IP del *Honeypot*

4.15. Escribir las opciones del comando "honeyd" , ver en la Figura 3.236.

```
# honeyd -d -i eth0 -f honeyd3.conf -p nmap.prints -x xprobe2.conf -a
nmap.assoc -0 pf.os -l /var/log/honeyd/honeyd3.log 192.168.1.20-192.168.1.22
```

Figura 3.236 Comandos de ejecución en Práctica No.3

4.16. Validación del demonio. Se debe considerar las condiciones del ingreso al equipo *Kali Linux* de la Práctica No.1.

4.17. Se debe verificar la creación del *Honeypot "Linux 2.4.20"* mediante el comando "*ping*".

ping 192.168.1.20

4.18. Visualizar la respuesta de *Honeyd* en ejecución en la Figura 3.237.

```
honeyd ["#proceso"]: Sending ICMP Echo Reply: 192.168.1.20 -> 192.168.1.30
```

Figura 3.237 Conexión de *Honeypot* 1 en red 3

4.19. Se procede a detener el comando "*ping*" en la máquina "atacante" mediante la opción "***Ctrl + c***" en el teclado.

4.20. Se procede a escanear los puertos TCP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-P0*".

nmap –send-ip -P0 192.168.1.20

4.21. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.238.

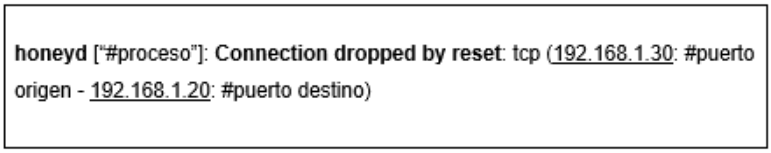


Figura 3.238 Escaneo TCP de *Honeypot* 1 en red 3

4.22. Se procede a escanear los puertos UDP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-sU*".

nmap –send-ip -sU 192.168.1.20

4.23. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.239.

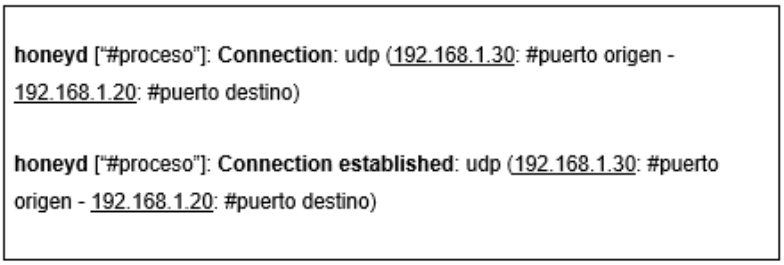


Figura 3.239 Escaneo UDP de *Honeypot* 1 en red 3

4.24. Se debe verificar la creación del *Honeypot "Apple MAC OS 7.1"* mediante el comando "*ping*".

ping 192.168.1.21

4.25. Visualizar la respuesta de *Honeyd* en ejecución en la Figura 3.240.

Figura 3.240 Conexión de *Honeypot* 2 en red 3

4.26. Se procede a detener el comando "*ping*" en la máquina "atacante" mediante la opción "***Ctrl + c***" en el teclado.

4.27. Se procede a escanear los puertos TCP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-P0*".

nmap –send-ip -P0 192.168.1.21

4.28. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.241.

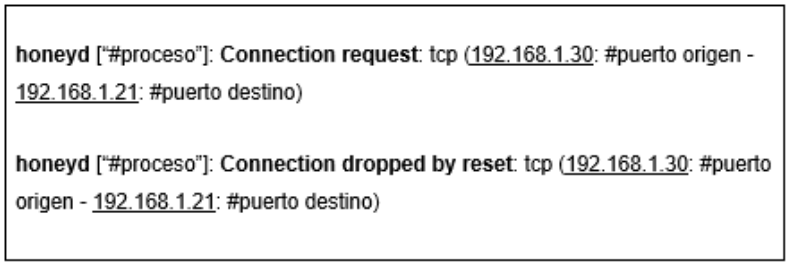


Figura 3.241 Escaneo TCP de *Honeypot* 2 en red 3

4.29. Se procede a escanear los puertos UDP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-sU*".

nmap –send-ip -sU 192.168.1.21

4.30. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.242.

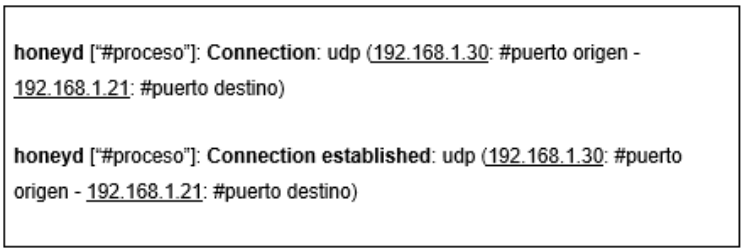


Figura 3.242 Escaneo UDP de *Honeypot* 2 en red 3

4.31. Se debe verificar la creación del *Honeypot "Linux Kernel 2.4.3 SMP (RedHat)"* mediante el comando "*ping*".

ping 192.168.1.22

4.32. Visualizar la falta de respuesta.

4.33. Se debe detener el comando "*ping*" en la máquina "atacante" mediante la opción "***Ctrl + c***" en el teclado.

4.34. Se opta por escanear los puertos TCP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-P0*".

nmap –send-ip -P0 192.168.1.22

4.35. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.243.

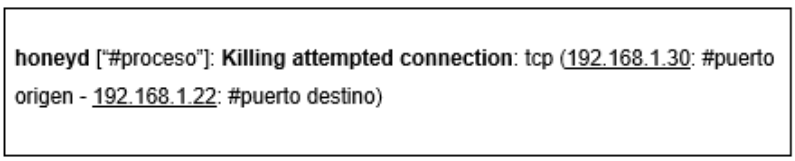


Figura 3.243 Escaneo TCP de *Honeypot* 3 en red 3

4.36. Se debe escanear los puertos UDP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-sU*".

nmap –send-ip -sU 192.168.1.2

4.37. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.244.

```
honeyd ["#proceso"]: Connection to closed port: udp (192.168.1.30: #puerto
origen - 192.168.1.22: #puerto destino)
```

Figura 3.244 Interacción UDP de *Honeypot* 3 en red 3

4.38. Se opta por detener el demonio de *Honeyd* mediante "***Ctrl + c***" en el teclado.

4.39. Visualizar el archivo de registro en el archivo "*honeyd3.log*" en el directorio "*/var/log/honeyd/"* mediante el comando "*vim*".

vim /var/log/honeyd/honeyd3.log

4.40. Visualizar los resultados en el archivo de registro.

- Análisis de *Honeypot* 1 "*Linux 2.4.20*"

TCP

Los resultados de puertos TCP se visualizan en la Figura 3.245.

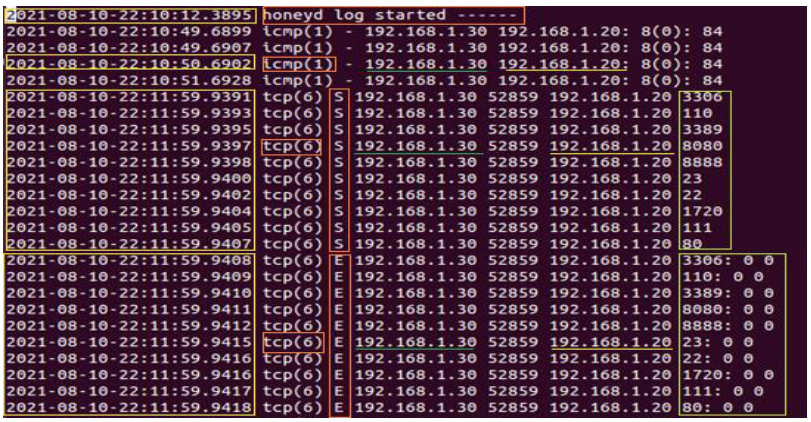


Figura 3.245 Análisis TCP de *Honeypot* 1 en red 3

1. Fecha y hora de inicialización del proceso de *Honeyd.*
2. Mensaje "ICMP" enviado entre *host.*
3. Inicio de conexión (S) en grupo de puertos TCP.
4. Finalización de conexión (E) en grupo de puertos TCP.

UDP

Los resultados de puertos UDP se visualizan en la Figura 3.246.

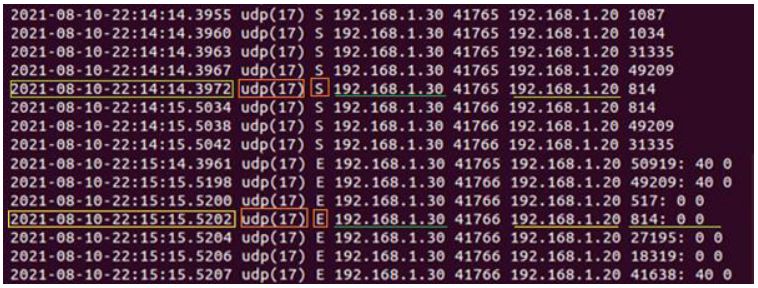


Figura 3.246 Análisis UDP de *Honeypot* 1 en red 3

1. Inicio de conexión (S) de puertos UDP.
2. Finalización de conexión (E) de puertos UDP.
3. Puertos por acción “open”.

– Análisis de *Honeypot* “*Apple MAC OS 7.1*”

TCP

Los resultados de puertos TCP se visualizan en la Figura 3.247.

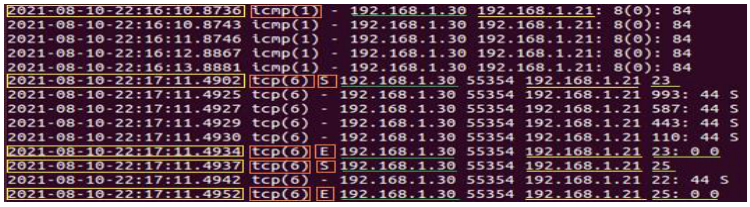


Figura 3.247 Análisis TCP de *Honeypot* 2 en red 3

1. Fecha y hora de proceso de *Honeyd.*
2. Mensaje “ICMP” enviado entre *host.*
3. Inicio de conexión (S) en grupo de puertos TCP 23 y 25.
4. Finalización de conexión (E) en grupo de puertos TCP 23 y 25.

UDP

Los resultados de puertos UDP se visualizan en la Figura 3.248.

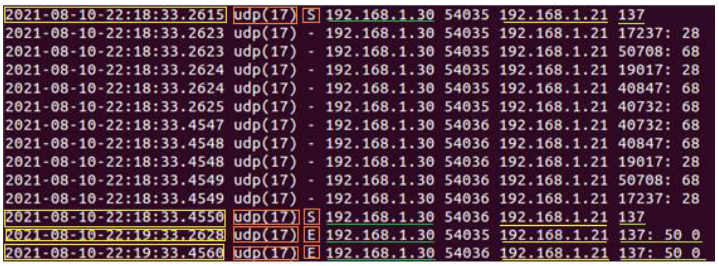


Figura 3.248 Análisis UDP de *Honeypot* 2 en red 3

1. Inicio de conexión (S) puerto UDP 137.
2. Finalización de conexión (E) puerto UDP 137.

– Análisis de *Honeypot* "*Linux Kernel 2.4.3 SMP (RedHat)*".

Los resultados de puertos TCP se visualizan en la Figura 3.249.

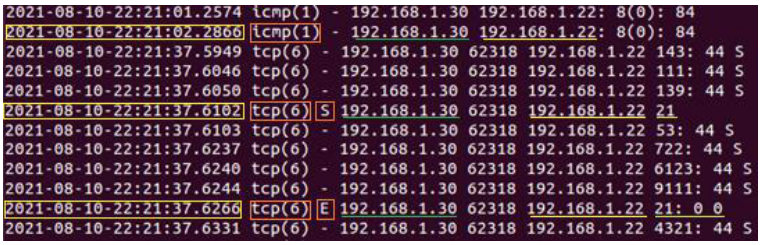


Figura 3.249 Análisis TCP de *Honeypot* 3 en red 3

1. Fecha y hora de proceso de *Honeyd.*
2. Sin mensaje "ICMP".
3. Inicio y finalización de conexión (E) en puerto TCP 21.

Los resultados de puertos TCP se visualizan en la Figura 3.250.

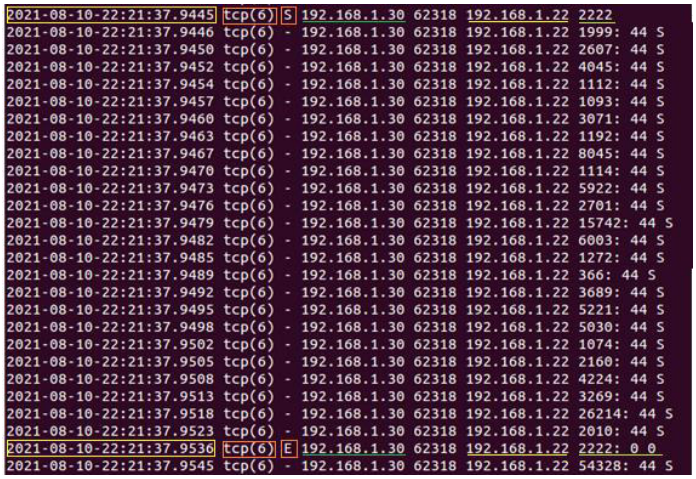


Figura 3.250 Análisis 2 TCP de *Honeypot* 3 en red 3

1. Inicio de conexión (S) en puerto TCP 2222.
2. Finalización de conexión (E) en puerto TCP 2222.

UDP

Los resultados de puertos UDP se visualizan en la Figura 3.251.

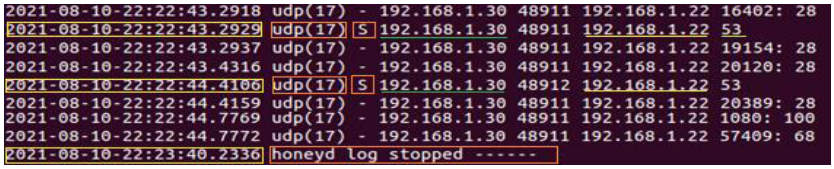


Figura 3.251 Análisis UDP de *Honeypot* 3 en red 3

1. Inicio de conexión (S) puerto UDP 53.
2. Finalización de conexión (E) puerto UDP 53.
3. Finalización del proceso de *Honeyd.*

5. INFORME

5.1. Investigar los métodos de protección por medio de puertos

5.2. Presentar el análisis de los resultados obtenidos del archivo de registro de información “*honeyd3.log*”.

PRÁCTICA 4

1. **TEMA:** Simulación de Servicios TCP
2. **OBJETIVOS**

2.1. OBJETIVO GENERAL

- Desarrollar un Sistema de Seguridad en base a la simulación de una red de servicios TCP mediante la aplicación de los "*Honeypots*".

2.2. OBJETIVOS ESPECÍFICOS

- Interactuar con Sistema Operativos *GNU/Linux Ubuntu*.
- Ejecutar una serie de comandos para obtención de un conjunto de *Honeypots*.
- Configurar simulación de Servicios TCP.
- Visualizar los resultados obtenidos.

3. **TRABAJO PREPARATORIO**

3.1. CUESTIONARIO

- Realizar un resumen hacer de un Servidor de Red.
- Realizar una representación gráfica de los servicios que se proyectan sobre los puertos de protocolos.
- Investigar las características más importantes de los siguientes servidores:
  - Servidor WEB
  - Servidor TELNET
  - Servidor FTP

4. **DESCRIPCIÓN DE ACTIVIDADES Y PROCEDIMIENTO DE LA PRÁCTICA**

4.1. Considerar las mismas condiciones de ingreso a la máquina virtual de *Honeyd_Ubuntu* de la Práctica No.1.

4.2. Crear una copia del archivo de configuración dentro del directorio "*honeyd*" con el nombre "*honeyd4conf*" mediante el comando "*cp*".

cp honeyd.conf /home/jose/honeyd/honeyd_kit-1.0c-a/honeyd4.conf

4.3. Visualizar y analizar la topología de Red *Honeypot* en la Figura 3.252.

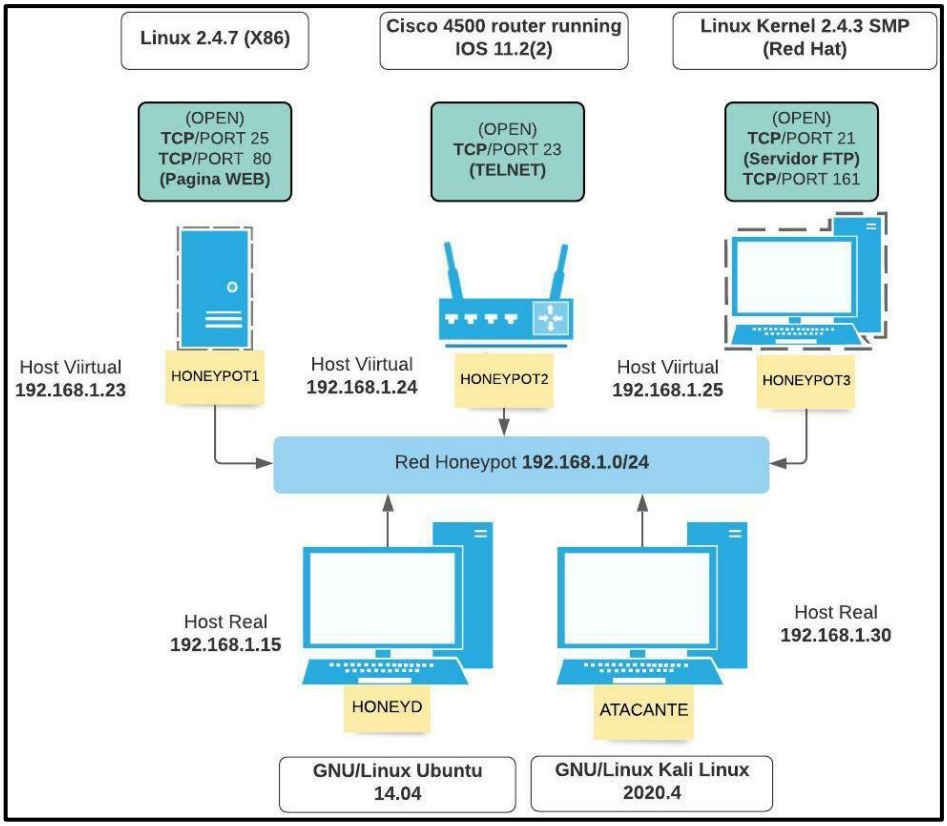


Figura 3.252 Topología de red 4

4.4. Elegir el sistema operativo o personalidad en el archivo "*Nmap.assoc*" mediante el comando "*vim".*

vim nmap.assoc

4.5. Seleccionar los sistemas operativos ***"Linux 2.4.7 (X86), Cisco 4500 router running IOS 11.2(2) y Linux Kernel 2.4.3 SMP (RedHat)*** ".

4.6. Se procede a establecer las direcciones IP de las personalidades, ver en la Tabla 3.24.

Tabla 3.24 Direcciones IP de Práctica No.4

Personalidad	Dirección IP
Linux 2.4.7 (X86)	192.168.1.23
Cisco 4500 router running IOS 11.2(2)	192.168.1.24
Linux Kernel 2.4.3 SMP (RedHat)	192.168.1.25

4.7. Ingreso al archivo de configuración "*honeyd4.conf*" mediante el comando "vim".

vim honeyd4.conf

4.8. Argumentos de configuración de "*Honeyd*" en la Tabla 3.25.

Tabla 3.25 Argumentos de configuración de Práctica No.4 [27] [32]

Argumento	Función
create	Creación de Personalidad o Sistema operativo
set	Establecimiento de acción
default	Acción por defecto
add	Agregar función o puerto
TCP/UDP	Protocolos
port	Puerto
open	Abierto
reset	*Reseteo*
block	Bloqueo
/scripts/	Directorio de *scripts* de simulación
$ipsrc $sport	Variables de dirección de IP origen- puerto origen
$ipdst $dport	Variables de dirección de IP destino- puerto destino
perl	Archivo tipo *script* denominado "*.pl*"
uptime	Tiempo de subida en segundos
bind	Enlace o dirección IP

4.9. Establecer los puertos de acuerdo a la Tabla 3.26.

Tabla 3.26 Puertos de personalidad de Práctica No.4 [27] [32]

Personalidad	Puertos		Acción
Linux 2.4.7 (X86)	80	HTTP	Página WEB
	25	SMTP	*Open*
Cisco 4500 router running IOS 11.2(2)	23	TELNET	Servidor TELNET
Linux Kernel 2.4.3 SMP (Red Hat)	21	FTP	Servidor FTP
	161	SNMP	*Open*

4.10. Implementar la configuración en el apartado final del archivo de configuración "***honeyd4.conf***", ver en la Figura 3.253.

```
### Plantilla de Honeypot "Linux"
create suse80
set suse80 personality "Linux 2.4.7 (X86)"
set suse80 default tcp action reset
set suse80 default udp action block
set suse80 default icmp action open
set suse80 uptime 79239
add suse80 tcp port 80 " /home/jose/honeyd/honeyd_kit-1.0c-
a/scripts/unix/linux/suse8.0/apache.sh $ipsrc $sport $ipdst $dport"
add suse80 tcp port 25 open

bind 192.168.1.23 suse80

### Plantilla de Honeypot "Cisco Router"
create router
set router personality "Cisco 4500 router running IOS 11.2(2)"
set router default tcp action reset
set router default udp action block
set router default icmp action open
set router uptime 79239
add router tcp port 23 "perl /home/jose/honeyd/honeyd_kit-1.0c-
a/scripts/router/cisco/router-telnet.pl"

bind 192.168.1.24 router

### Plantilla de Honeypot "Kernel"
create kernet
set kernet personality "Linux Kernel 2.4.3 SMP (RedHat)"
set kernet default tcp action reset
set kernet default udp action block
set kernet default icmp action open
set kernet uptime 79239
add kernet tcp port 21 "/home/jose/honeyd/honeyd_kit-1.0c-
a/scripts/unix/linux/ftp.sh"
add kernet tcp port 161 open

bind 192.168.1.25 kernet
```

Figura 3.253 Configuración de Práctica No.4

4.11. Guardar el archivo de configuración mediante la tecla "***Esc***" y escribir a opción "***:wq***".

4.12. Arranque *Honeyd.* Se inicia con el arranque del demonio "*arpd*" mediante el *script* "*start-arpd.sh*" con el comando "*./*".

./start-arpd.sh

4.13. Arranque del demonio de "*Honeyd*" mediante el *script* "*star-honeyd.sh*" con el comando "*./*".

./star-honeyd.sh

4.14. Argumentos de arranque de *Honeyd,* ver en la Tabla 3.27.

Tabla 3.27 Argumentos de arranque de red 4 [27] [32]

Opción	Función
honeyd	Arranque de *Honeyd*
-d	Levantamiento del Demonio
-i	Tarjeta de Red "eth0"
-f	Archivo de configuración- ruta "*Honeyd4.conf*"
-p	Archivo "*Nmap.prints*" conjunto de personalidades-ruta
-x	Archivo "*xprobe2.conf*" sonda-lector de personalidades-ruta
-a	Archivo "*Nmap.assoc*" personalidades-sonda-ruta
-0	Archivo "*pf.os*" lector de personalidades pasivas-ruta
-l	Archivo de Registro de Información "*Honeyd4.log*"
192.168.1.23-192.168.1.25	Dirección IP del *Honeypot*

4.15. Escribir las opciones del comando "honeyd", ver en la Figura 3.254.

```
# honeyd -d -i eth0 -f honeyd4.conf -p nmap.prints -x xprobe2.conf -a
nmap.assoc -0 pf.os -l /var/log/honeyd/honeyd4.log 192.168.1.23-
192.168.1.25
```

Figura 3.254 Comandos de ejecución de Práctica No.4

4.16. Validación de demonio. Se debe considerar las mismas condiciones del ingreso al equipo *Kali Linux* de la Práctica No.1.

4.17. Se debe verificar la creación del *Honeypot "Linux 2.4.7 (X86)""* mediante el comando "*ping*".

ping 192.168.1.23

4.18. Visualizar la respuesta de *Honeyd* en ejecución en la Figura 3.255.

```
honeyd ["#proceso"]: Sending ICMP Echo Reply: 192.168.1.23 -> 192.168.1.30
```

Figura 3.255 Conexión de *Honeypot* 1 en red 4

4.19. Se procede a detener el comando "*ping*" en la máquina "atacante" mediante la opción "***Ctrl + c***" en el teclado.

4.20. Se procede a escanear los puertos TCP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-P0*".

nmap –send-ip -P0 192.168.1.23

4.21. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.256.

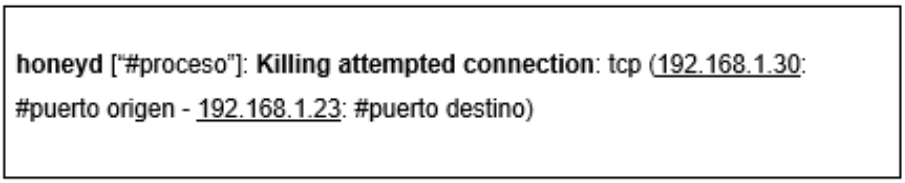


Figura 3.256 Escaneo TCP en *Honeypot* 1 en red 4

4.22. Validar la página web en el equipo *Honeyd* sobre el "atacante" en la dirección 192.168.1.23, ver en la Figura 3.257.

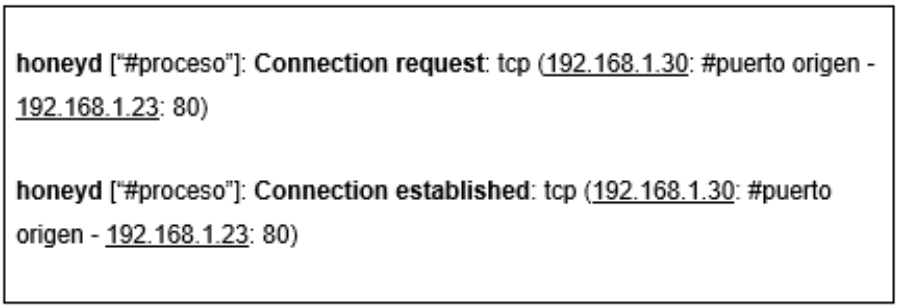


Figura 3.257 Validación de servicio de *Honeypot* 1 en red 4

4.23. Se debe verificar la creación del *Honeypot "Cisco 4500 router running IOS 11.2(2)"* mediante el comando "*ping*".

ping 192.168.1.21

4.24. Visualizar la respuesta de *Honeyd* en ejecución en la Figura 3.258.

Figura 3.258 Conexión de *Honeypot* 2 en red 4

4.25. Se opta por detener el comando "*ping*" en la máquina "atacante" mediante la opción "***Ctrl + c***" en el teclado.

4.26. Se debe escanear los puertos TCP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-P0*".

nmap –send-ip -P0 192.168.1.24

4.27. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.259.

Figura 3.259 Escaneo TCP en *Honeypot* 2 en red 4

4.28. Validar el servidor Telnet en el "atacante" en la dirección 192.168.1.24 mediante el comando "*telnet 192.168.1.24*".

4.29. Se debe ingresar las credenciales y saldrá por defecto "Acceso Denegado"

4.30. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.260.

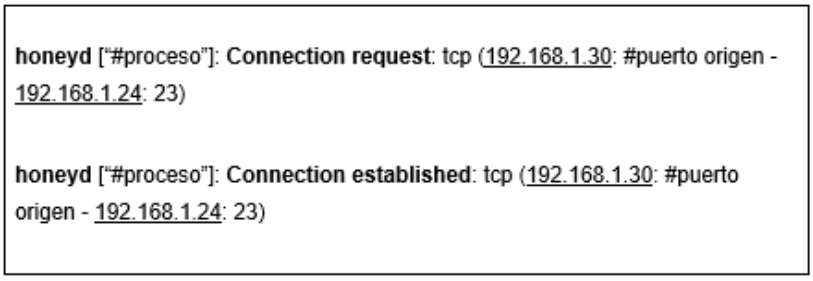


Figura 3.260 Validación de servicio de *Honeypot* 2 en red 4

4.31. Se opta por verificar la creación del *Honeypot* "*Linux Kernel 2.4.3 SMP (RedHat)"* mediante el comando "*ping*".

ping 192.168.1.21

4.32. Visualizar la respuesta de *Honeyd* en ejecución en la Figura 3.261.

Figura 3.261 Conexión de *Honeypot* 3 en red 4

4.33. Se debe detener el comando "*ping*" en la máquina "atacante" mediante la opción "***Ctrl + c***" en el teclado.

4.34. Se opta por escanear los puertos TCP en el equipo *Kali Linux* mediante el comando "*Nmap*", las opciones *"—send-ip*" y "*-P0*".

nmap –send-ip -P0 192.168.1.25

4.35. Se presentan los resultados del equipo *Honeyd*, ver en la Figura 3.262.

honeyd ["#proceso"]: **Killing attempted connection**: tcp (192.168.1.30: #puerto origen - 192.168.1.25: #puerto destino)

Figura 3.262 Escaneo TCP en *Honeypot* 3 en red 4

4.36. Validar el servidor FTP en el "atacante" en la dirección 192.168.1.25 mediante el comando" FTP *192.168.1.25*".

4.37. Ingresar las credenciales y comandos del servidor. Se produce fallo debido a que es un señuelo no un servicio real como tal. *Honeyd* notifica, ver en la Figura 3.263.

honeyd ["#proceso"]: **Connection request**: tcp (192.168.1.30: #puerto origen - 192.168.1.25: 21)

honeyd ["#proceso"]: **Connection established**: tcp (192.168.1.30: #puerto origen - 192.168.1.25: 21)

Figura 3.263 Validación de servicio de *Honeypot* 3 en red 4

4.38. Se debe a detener el demonio de *Honeyd* mediante "***Ctrl + c***" en el teclado.

4.39. Visualizar el archivo de registro en el archivo "*honeyd4.log*" en el directorio "*/var/log/honeyd/"* mediante el comando "*vim*" .

vim /var/log/honeyd/honeyd4.log

4.40. Visualizar los resultados en el archivo de registro

- Análisis de *Honeypot* "*Linux 2.4.7 (X86)"*

  El resultado de conexión y puertos TCP se visualiza en las Figura 3.264 y Figura 3.265.

```
2021-08-06-10:45:52.4737 honeyd log started ------
2021-08-06-10:46:05.6403 icmp(1) - 192.168.1.30 192.168.1.23: 8(0): 84
2021-08-06-10:46:06.6422 icmp(1) - 192.168.1.30 192.168.1.23: 8(0): 84
2021-08-06-10:46:07.6438 icmp(1) - 192.168.1.30 192.168.1.23: 8(0): 84
2021-08-06-10:46:08.6466 icmp(1) - 192.168.1.30 192.168.1.23: 8(0): 84
2021-08-06-10:47:23.6171 tcp(6) - 192.168.1.30 50369 192.168.1.23 3306: 44 S
2021-08-06-10:47:23.6174 tcp(6) - 192.168.1.30 50369 192.168.1.23 587: 44 S
2021-08-06-10:47:23.6179 tcp(6) - 192.168.1.30 50369 192.168.1.23 21: 44 S
2021-08-06-10:47:23.6180 tcp(6) - 192.168.1.30 50369 192.168.1.23 143: 44 S
2021-08-06-10:47:23.6182 tcp(6) - 192.168.1.30 50369 192.168.1.23 113: 44 S
2021-08-06-10:47:23.6185 tcp(6) S 192.168.1.30 50369 192.168.1.23 25
2021-08-06-10:47:23.6185 tcp(6) - 192.168.1.30 50369 192.168.1.23 5900: 44 S
2021-08-06-10:47:23.6189 tcp(6) - 192.168.1.30 50369 192.168.1.23 554: 44 S
2021-08-06-10:47:23.6191 tcp(6) - 192.168.1.30 50369 192.168.1.23 8080: 44 S
2021-08-06-10:47:23.6193 tcp(6) - 192.168.1.30 50369 192.168.1.23 995: 44 S
2021-08-06-10:47:23.6197 tcp(6) E 192.168.1.30 50369 192.168.1.23 25: 0 0
2021-08-06-10:47:23.6202 tcp(6) - 192.168.1.30 50369 192.168.1.23 139: 44 S
```

Figura 3.264 Análisis TCP en *Honeypot* 1 de red 4

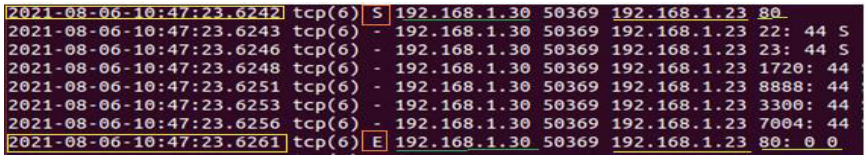


Figura 3.265 Análisis 2 TCP en *Honeypot* 1 de red 4

1. Fecha y hora de Inicialización del proceso de *Honeyd*.
2. Mensaje “ICMP” enviado entre *host*.
3. Inicio de conexión (S) en grupo de puertos TCP.
4. Finalización de conexión (E) en grupo de puertos TCP.

Análisis de página Web (Apache)

El resultado de servicio se visualiza en la Figura 3.266.

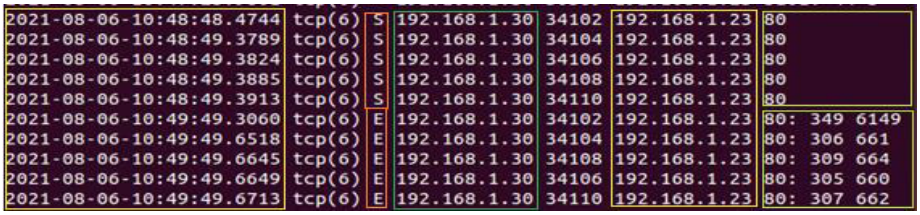


Figura 3.266 Análisis TCP de servicio de *Honeypot* 1 en red 4

1. Establecimiento de conexión (S) de puerto TCP 80.
2. Finalización de conexión (E) de puerto TCP 80.
3. Valor de paquetes enviados de fuente a destino.

– Análisis de *Honeypot* “*Cisco 4500 router running IOS 11.2(2)*"
El resultado de conexión y puertos TCP se visualiza en las Figura 3.267 y Figura 3.268.

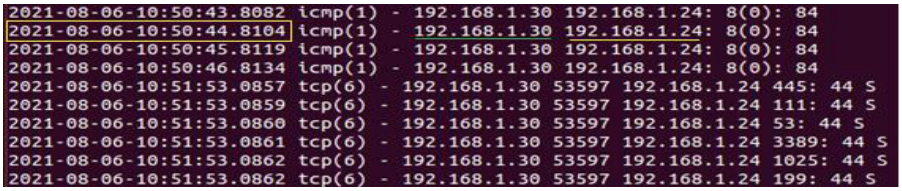


Figura 3.267 Análisis TCP en *Honeypot* 2 de red 4

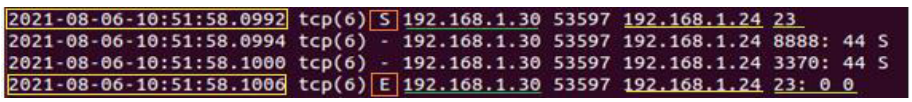


Figura 3.268 Análisis 2 TCP en *Honeypot* 2 de red 4

1. Fecha y hora de proceso de *Honeyd*.
2. Mensaje “ICMP” enviado entre *host*.
3. Inicio de conexión (S) en puerto TCP 23.
4. Finalización de conexión (E) en puerto TCP 23.

Análisis de servidor Telnet

El resultado de servicio se visualiza en la Figura 3.269.

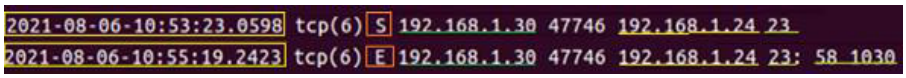


Figura 3.269 Análisis TCP de servicio de *Honeypot* 2 en red 4

1. Establecimiento de conexión (S) de puerto TCP 23.
2. Finalización de conexión (E) de puerto TCP 23.
3. Valor de paquetes enviados de fuente a destino.

– Análisis de *Honeypot* “*Linux Kernel 2.4.3 SMP (RedHat)*”

El resultado de conexión y puertos TCP se visualiza en la Figura 3.270.

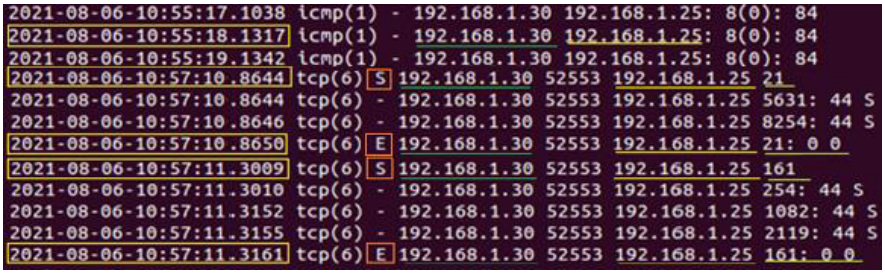


Figura 3.270 Análisis TCP en *Honeypot* 3 de red 4

1. Fecha y hora de proceso de *Honeyd.*
2. Sin mensaje “ICMP”.
3. Inicio de conexión (S) en el grupo de puertos TCP 21 y 161.
4. Finalización de conexión (E) en el grupo de puertos TCP 21 y 161.

Análisis de servicio FTP

El resultado de servicio se visualiza en la Figura 3.271.

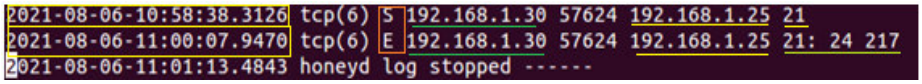


Figura 3.271 Análisis TCP de servicio en *Honeypot* 3 en red 4

1. Inicio de conexión (S) en puerto TCP 21.
2. Finalización de conexión (E) en puerto TCP 21.
3. Valor de paquetes enviados de fuente a destino.
4. Finalización del proceso de *Honeyd*.

5. INFORME

5.1. Analizar el estado y funciones de servicios TCP del conjunto de *Honeypots* empleados.

5.2. Presentar el análisis de los resultados obtenidos del archivo de registro de información "*honeyd4.log*".

3.5 Verificación de funcionamiento

El desarrollo de la verificación se realiza en base al proceso de implementación del módulo sobre el Laboratorio Marcelo Dávila. Por lo cual, se maneja el procesamiento de la prueba y análisis de la importación de máquinas.

Importación de máquinas

1. La importación de máquinas se describe como el proceso del proyecto sobre la implementación del módulo en el Laboratorio Marcelo Dávila, con el fin de dar a conocer su funcionamiento.
2. Se procede a importar las máquinas desde repositorio digital de EPN otorgado por el correo institucional. Además, se encarga de cargar las máquinas virtuales sobre los ordenadores del Laboratorio Marcelo Dávila, establecidos por el jefe de laboratorios de TIC´s de la ESFOT.
3. Se debe descargar del repositorio digital del correo institucional, ver en la Figura 3.272.

Figura 3.272 Repositorio de personal de máquinas

4. Se opta por seleccionar en la opción de “Importar” sobre el programa *Oracle VirtualBox*, ver en la Figura 3.273.

Figura 3.273 Importación de máquinas

.

5. Se debe seleccionar la distribución de *Ubuntu*, ver la Figura 3.274.

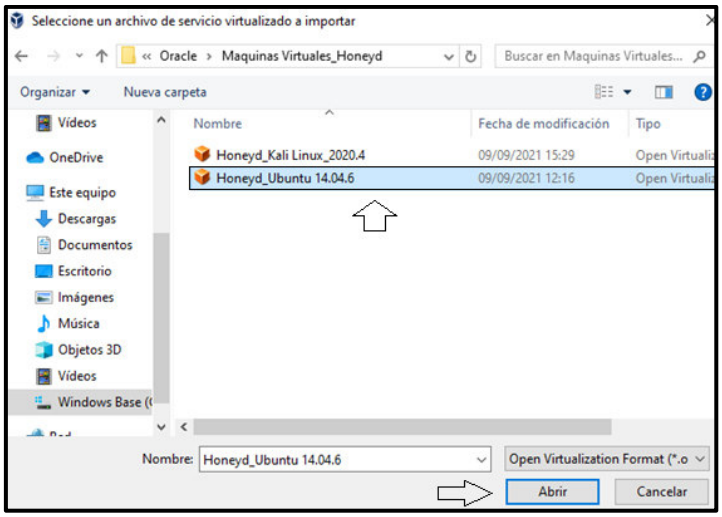


Figura 3.274 Selección de distribución *Ubuntu*

6. Se opta por cargar la distribución *Ubuntu* en base a las características configuradas, ver en la Figura 3.275.

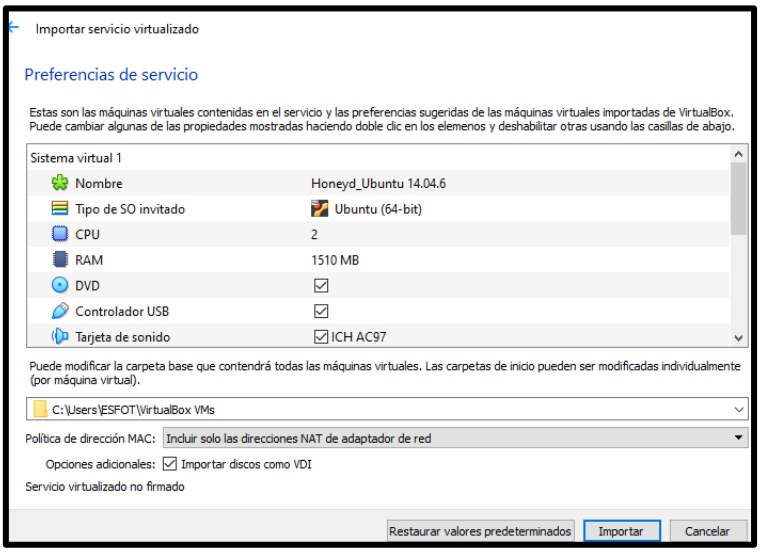


Figura 3.275 Carga de características del sistema *Ubuntu*

7. Se establece la máquina virtual *Ubuntu*, ver en la Figura 3.276.

Figura 3.276 Disposición de máquina virtual *Ubuntu*

8. Se debe seleccionar la distribución de *Kali Linux*, ver la Figura 3.277.

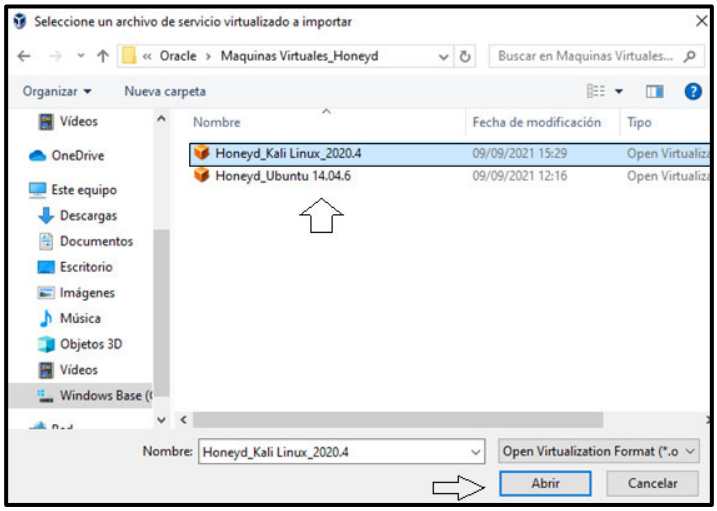


Figura 3.277 Selección de distribución *Kali Linux*

9. Se opta por cargar la distribución *Kali Linux* en base a las características configuradas, ver en la Figura 3.278.

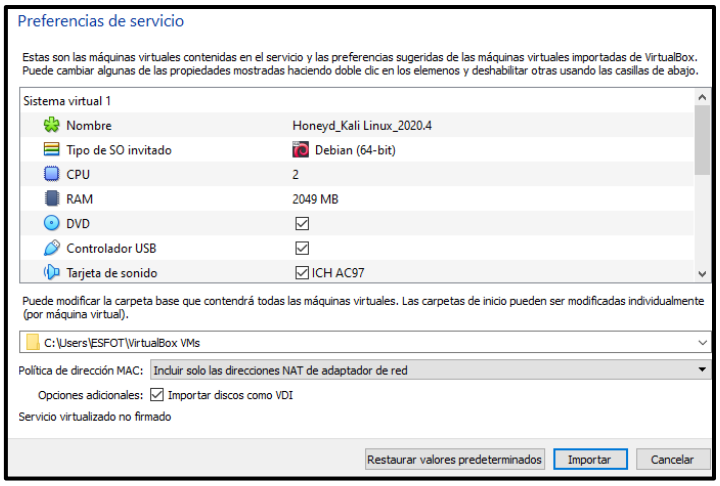


Figura 3.278 Carga de características del sistema *Kali Linux*

10. Se establece la máquina virtual *Kali Linux*, ver en la Figura 3.279.

Figura 3.279 Disposición de máquina virtual *Kali Linux*.

11. Se debe ejecutar la máquina virtual *Ubuntu*, ver en la Figura 3.280.

Figura 3.280 Ejecución del sistema *Ubuntu* en ordenador

12. Se opta por ejecutar la máquina virtual *Kali Linux*, ver en la Figura 3.281

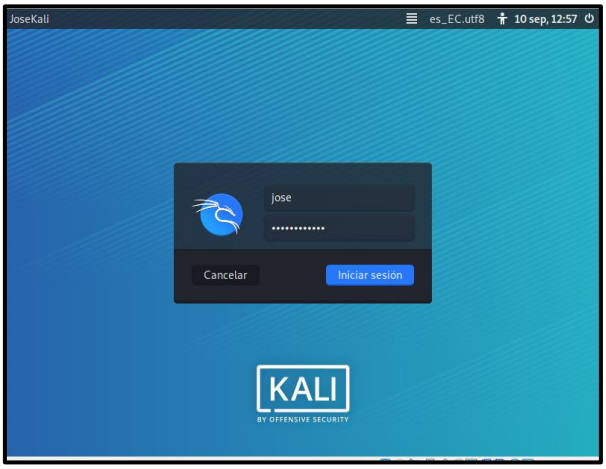


Figura 3.281 Ejecución del sistema *Kali Linux* en ordenador

13. Representación de la máquina virtual de *Ubuntu* sobre 20 ordenadores del laboratorio, ver en la Figura 3.282.

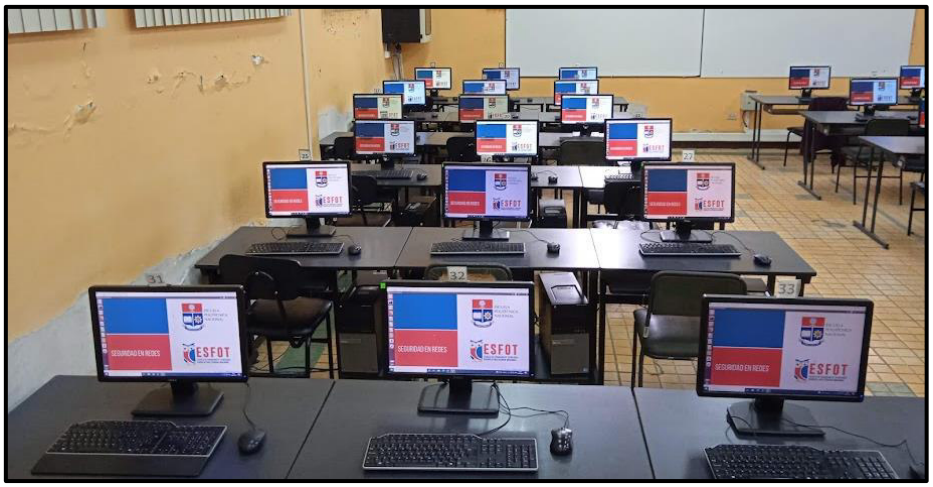


Figura 3.282 Disposición de *Ubuntu* instalado en 20 ordenadores

14. Representación de la máquina virtual de *Ubuntu* sobre 17 ordenadores del laboratorio, ver en la Figura 3.283.

Figura 3.283 Disposición de *Ubuntu* instalado en 17 ordenadores

15. Representación global de la máquina virtual de *Ubuntu* sobre el laboratorio, ver en la Figura 3.284.

Figura 3.284 Disposición global de *Ubuntu*

16. Representación de la máquina virtual de *Kali Linux* sobre 20 ordenadores del laboratorio, ver en la Figura 3.285.

Figura 3.285 Disposición de *Kali Linux* instalado en 20 ordenadores

17. Representación de la máquina virtual de *Kali Linux* sobre 17 ordenadores del laboratorio, ver en la Figura 3.286.

Figura 3.286 Disposición de *Kali Linux* instalado en 17 ordenadores

18. Representación global de la máquina virtual de *Kali Linux* sobre el laboratorio, ver en la Figura 3.287.

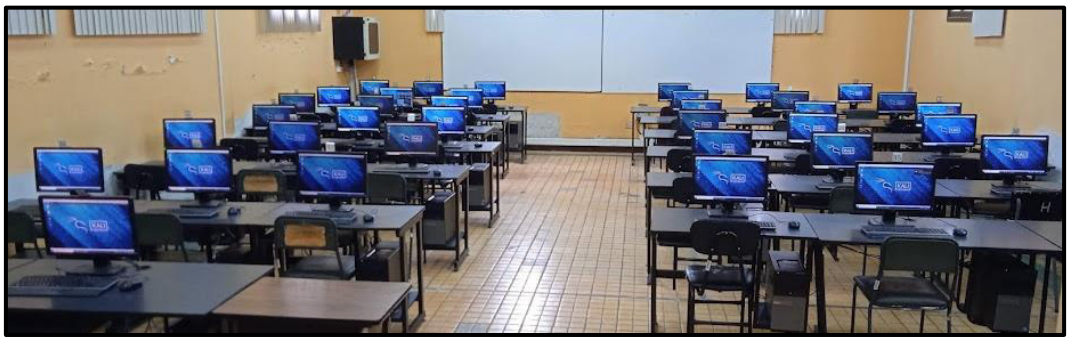


Figura 3.287 Disposición global de *Kali Linux*

19. Se describe el entorno para la utilización de los estudiantes de la asignatura de Seguridad en Redes en la disposición de 37 equipos.
20. El módulo virtual se mantiene estable por las características de las máquinas virtuales. Sin embargo, se puede realizar la actualización de los repositorios empleando los comandos "*sudo update*" y "*sudo upgrade*" en la terminal de los sistemas operativos.

Validación de funcionamiento

La verificación de funcionamiento del módulo virtual se realiza en base a un video demostrativo de aplicación de cada una de las prácticas. A continuación, se presentan los códigos QR de acceso a cada video demostrativo:

- **Práctica No.1.-** El enlace QR correspondiente al video demostrativo de la Práctica No.1, se visualiza en la Figura 3.288.

Figura 3.288 Práctica No.1 Introducción a los *Honeypots*

- **Práctica No.2.-** El enlace QR correspondiente al video demostrativo de la Práctica No.2, se visualiza en la Figura 3.289.

Figura 3.289 Práctica No.2 Validación de Puertos TCP y UDP

- **Práctica No.3.-** El enlace QR correspondiente al video demostrativo de la Práctica No.1, se visualiza en la Figura 3.290.

Figura 3.290 Práctica No.3 Acciones de Protocolos TCP, UDP e ICMP

- **Práctica No.4.-** El enlace QR correspondiente al video demostrativo de la Práctica No.1, se visualiza en la Figura 3.291.

Figura 3.291 Práctica No.4 Servicios de Puertos TCP

4 CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- El programa de virtualización *Oracle VirtualBox* ayudó en el desarrollo en cuanto a la instalación y configuración de las distribuciones de los sistemas operativos basados en *GNU/Linux*. Dichos sistemas servirán para el manejo adecuado del módulo virtual sobre los ordenadores del Laboratorio Sala Marcelo Dávila en la asignatura de Seguridad en Redes.
- El empleo de imágenes ISO de sistemas operativos garantiza el desarrollo de entornos de escritorio sobre esquemas virtuales, otorgando una eficiencia de los recursos físicos sobre cualquier tipo de ordenador o servidor.
- El uso de la Licencia General Pública (GPL) permite la utilización y manejo de programas de código abierto en áreas de desarrollo académico. En tal virtud, las configuraciones empleadas sobre estos sistemas establecen mayor flexibilidad de uso y se encuentran disponibles gratuitamente.
- El proceso de instalación del paquete del software *Honeyd* permite manejar con mayor destreza la creación de paquetes en los sistemas operativos *GNU/Linux*, porque posee diferentes procesamientos en cuanto a librerías y herramientas de construcción de *software*.
- El programa *Honeyd* realiza la creación de equipos virtuales o imitación de sistemas operativos en un entorno simulado, debido a que posee características relacionadas a la interacción de nivel bajo con atacantes, investigación de ataques y recolección de información en el área de la Seguridad Informática.
- Las personalidades o nombres de equipos desarrollados por *Honeyd* sirven para ajustar las características de simulación; es decir; con ello se ajusta el señuelo o trampa por parte del sistema y se protege al equipo principal.

- La arquitectura de *Honeyd* garantiza un procesamiento simple debido al manejo de un diagrama de bloques, donde los paquetes se enrutan de manera continua sobre los protocolos ICMP, TCP y UDP, los cuales realizan operaciones de servicio o establecimiento de conexiones entre equipos en base a las direcciones IP.
- El archivo "*Honeyd.conf*" ayudó en el desarrollo de los equipos virtuales o *Honeypots* en cuanto a la configuración, programación y ejecución de líneas de comando del entorno de *Honeyd*. Lo cual permite el aprendizaje y manipulación de los archivos del tipo texto o "*scripts*".
- La creación de las hojas guías para el desarrollo de las prácticas de laboratorio garantizan el correcto funcionamiento del módulo en base a la descripción detallada de cada procedimiento.
- La conversión de páginas de Internet en códigos QR sirve para la convergencia de tecnología en el manejo de aplicativos y visualización de procesos ubicados en la red.

4.2 Recomendaciones

- Se recomienda realizar la instalación del sistema de virtualización de *Oracle VirtualBox* sobre ordenadores con requerimientos mínimos; es decir; con recursos físicos disponibles como el procesador Intel (o compatibles), memoria RAM de 4 (GB) y disco duro de 32 (GB).
- Se sugiere utilizar versiones estables de las distribuciones del sistema operativo *GNU/Linux*, a fin de mantener un sistema compatible con cualquier tipo de programas o paquetes, para mantener un procesamiento normal del sistema.
- Se debe considerar la descarga de archivos y paquetes en repositorios oficiales; una mala descarga puede ocasionar error o inconvenientes sobre el proceso de instalación.
- En la base de datos del archivo "*Nmap.assoc*", correspondiente a las personalidades de *Honeyd* se recomienda seleccionar la descripción completa del nombre del equipo, con el fin de evitar errores de complicación en ejecución del servicio.
- Para la verificación de los adaptadores de red se recomienda ajustar las tarjetas de red de acuerdo a la numeración continua (1 y 2), en orden establecido. Se sugiere realizar el ingreso a los escritorios de las máquinas virtuales y configurar el ícono de red.

- Se recomienda establecer las líneas de comando en el archivo de configuración "*Honeyd.conf*" en el apartado final del archivo, con el fin que pueda simular la personalidad con todos los argumentos, debido a que si se ubica en una posición diferente genera o simula tan solo el nombre de la etiqueta.
- Se recomienda inicializar la máquina virtual de *Kali Linux* una vez configurado el archivo de *Honeyd* en la máquina virtual *Ubuntu*, con el fin de realizar las pruebas de funcionamiento en el momento que el proceso *Honeyd* se ejecute.
- En el proceso de escaneo de puertos se recomienda tomar en cuenta un tiempo de consideración de alrededor 30 segundos, debido a que los comandos de *Nmap* toman el tiempo requerido para entregar la información del escaneo realizado.
- Para el proceso de validación de servicio se recomienda realizar el proceso antes del cierre o el tiempo de ejecución, con el fin de evitar respuestas erróneas o resultados no esperados.

5 REFERENCIAS BIBLIOGRÁFICAS

[1] N. Poggi, «24 Estadísticas de Seguridad Informática que Importan en el 2021,» 19 Mayo 2021. [En línea]. Available: https://preyproject.com/blog/es/24-estadisticas-seguridad-informatica-2019/. [Último acceso: 12 Julio 2021].

[2] B. Woodbury, «Detección de dispositivos de red y administración de vulnerabilidades,» 19 Noviembre 2021. [En línea]. Available: https://docs.microsoft.com/es-es/microsoft-365/security/defender-endpoint/network-devices?view=o365-worldwide. [Último acceso: 22 Noviembre 2021].

[3] R. García, «Escáner De Puertos: ¿Qué es?, ¿Qué hace? y ¿Por qué los hackers lo aman?,» 7 Junio 2020. [En línea]. Available: https://www.hackbysecurity.com/blog/escaneado-puertos. [Último acceso: 1 Julio 2020].

[4] L. Calvo, «¿Qué es un Honeypot? y ¿Cómo usarlo?,» 4 Noviembre 2021. [En línea]. Available: https://es.godaddy.com/blog/que-es-un-honeypot-y-como-usarlo-en-beneficio-de-tu-negocio/. [Último acceso: 4 Noviembre 2021].

[5] L. Elena, «Vitualización ¿Qué es? ¿Para qué sirve? Ventajas,» 05 Mayo 2021. [En línea]. Available: https://openwebinars.net/blog/virtualizacion-que-es-para-que-sirve-y-ventajas/. [Último acceso: 18 Agosto 2021].

[6] IBM Cloud Education, «Virtualización,» 19 Junio 2019. [En línea]. Available: https://www.ibm.com/es-es/cloud/learn/virtualization-a-complete-guide. [Último acceso: 19 Agosto 2021].

[7] L. Sandoval, «¿Qué es la Virtualización de Sistemas Operativos?,» 16 Diciembre 2020. [En línea]. Available: https://ventanainformatica.com/software/virtualizacion-de-sistemas-operativos/#:~:text=La%20virtualizaci%C3%B3n%20de%20Sistemas%20Operati

vos%20funciona%20a%20trav%C3%A9s,operativos%20que%20se%20instalan%20bajo%20un%20solo%20hardware.. [Último acceso: 2 Noviembre 2021].

[8] IBM Cloud Education, «Máquinas virtuales,» 20 Junio 2019. [En línea]. Available: https://www.ibm.com/es-es/cloud/learn/virtual-machines. [Último acceso: 2 Noviembre 2021].

[9] R. Iván, «Máquinas virtuales; ¿Qué son?, ¿Cómo funcionan?, ¿Cómo utilizarlas?,» 25 Julio 2016. [En línea]. Available: https://www.xataka.com/especiales/maquinas-virtuales-que-son-como-funcionan-y-como-utilizarlas. [Último acceso: 20 Agosto 2021].

[10] Ingeniero Binario, «VirtualBox y Máquinas Virtuales,» 2021. [En línea]. Available: https://www.ingenierobinario.com/ubuntuvirtualbox/. [Último acceso: 2 Noviembre 2021].

[11] A. E. Ortiz, «¿Qué es un Hipervisor?,» 10 Septiembre 2019. [En línea]. Available: https://www.hostdime.com.ar/blog/que-es-un-hipervisor-tipos-de-hipervisores-1-y-2/. [Último acceso: 2 Noviembre 2021].

[12] E. G. Galán, «Virtualización basada en hipervisor,» 2 Noviembre 2021. [En línea]. Available: https://infseg.com/index.php/2016/02/07/comencemos-con-hypervirtualizacion/. [Último acceso: 10 Novimbre 2021].

[13] VIRTUALBOX, «VirtualBox,» [En línea]. Available: https://www.virtualbox.org/. [Último acceso: 02 Julio 2021].

[14] M. Navas, «¿Qué es Unix?,» 10 Noviembre 2016. [En línea]. Available: https://www.profesionalreview.com/2016/11/10/que-es-unix/. [Último acceso: 1 Noviembre 2021].

[15] J. M. Uriarte, «"Unix",» 2019 Diciembre 20. [En línea]. Available: https://www.caracteristicas.co/unix/. [Último acceso: 2 Noviembre 2021].

[16] H. Delgado, «El sistema de ficheros de UNIX,» 8 Junio 2021. [En línea]. Available: https://disenowebakus.net/el-sistema-de-ficheros-de-unix.php. [Último acceso: 2 Noviembre 2021].

[17] D. García, «¿Qué es GNU/Linux?,» 14 Junio 2019. [En línea]. Available: https://www.xn--linuxenespaol-skb.com/ayuda/que-es-gnu-linux/. [Último acceso: 1 Noviembre 2021].

[18] Y. Fernández, «¿Cuál es la diferencia entre Linux y GNU/Linux?,» 27 Septiembre 2018. [En línea]. Available: https://www.xataka.com/basics/cual-es-la-diferencia-entre-linux-y-gnu-linux. [Último acceso: 2 Novimbre 2021].

[19] L. Machado, «Ubuntu, la filosofía que ayuda a vivir mejor,» 28 Junio 2021. [En línea]. Available: https://elpais.com/planeta-futuro/2021-06-29/ubuntu-la-filosofia-que-ayuda-a-vivir-mejor.html. [Último acceso: 2 Noviembre 2021].

[20] Hostingplus, «¿Qué es?, Características y ¿Comó funciona?,» 22 Junio 2020. [En línea]. Available: https://www.hostingplus.pe/blog/ubuntu-que-es-caracteristicas-y-como-funciona/. [Último acceso: 2 Noviembre 2021].

[21] R. Altube, «Kali Linux: ¿Qué es? y Características principales,» 5 Noviembre 2021. [En línea]. Available: https://openwebinars.net/blog/kali-linux-que-es-y-caracteristicas-principales/. [Último acceso: 10 Noviembre 2021].

[22] A. Caballero, «Principales Características de Kali Linux,» 21 Noviembre 2019. [En línea]. Available: http://www.reydes.com/d/?q=Principales_Caracteristicas_de_Kali_Linux. [Último acceso: 2 Noviembre 2021].

[23] A. Warburton, «https://www.welivesecurity.com/la-es/2020/07/31/que-es-honeypot-como-implementarlo-nuestra-red/,» 31 Julio 2020. [En línea]. Available: https://www.welivesecurity.com/la-es/2020/07/31/que-es-honeypot-como-implementarlo-nuestra-red/. [Último acceso: 3 Noviembre 2021].

[24] O. Espinosa, «¿Qué es? y ¿Para qué sirve un Honeypot?,» 27 Julio 2021. [En línea]. Available: https://www.redeszone.net/tutoriales/seguridad/que-es-honeypot/. [Último acceso: 3 Noviembre 2021].

[25] J. Olano, «Monitorización de Honeypots,» 7 Julio 2021. [En línea]. Available: https://pandorafms.com/blog/es/honey-pots/. [Último acceso: 2 Novimbre 2021].

[26] Instituto Nacional de Ciberseguridad - ESPAÑA, «Guía de implantación de un Honeypot Industrial,» Octubre 2019. [En línea]. Available: https://www.incibe-

cert.es/sites/default/files/contenidos/guias/doc/incibe-cert_guia_implantacion_honeypot_industrial.pdf. [Último acceso: 3 Noviembre 2021].

[27] N. Provos, «Desarrollos de Honeyd Virtual Honeypot,» 27 Mayo 2007. [En línea]. Available: http://www.honeyd.org/faq.php. [Último acceso: 3 Noviembre 2021].

[28] T. Holz y N. Provos, Virtual Honeypots: From Botnet Tracking to Intrusion Detection, Reading, Massachusetts-USA: Addison Wesley Professional, 2007.

[29] S. D. Luz, «Realiza escaneos de puertos con Nmap a cualquier servidor o sistema,» 10 Junio 2021. [En línea]. Available: https://www.redeszone.net/tutoriales/configuracion-puertos/nmap-escanear-puertos-comandos/. [Último acceso: 4 Noviembre 2021].

[30] W. Nacimba, «SALA DE INTERNET "MARCELO DÁVILA",» 2021. [En línea]. Available: https://esfot.epn.edu.ec/index.php/esfot/436-lab-smd. [Último acceso: 4 Noviembre 2021].

[31] F. García, «Requisitos de Instalación de VirtualBox,» 15 Diciembre 2021. [En línea]. Available: https://esenciavital.es/informatica/requisitos-para-instalar-virtualbox/. [Último acceso: 16 Diciembre 2021].

[32] Niel Provos, «Preguntas frecuentes sobre Honeyd,» 27 Mayo 2007. [En línea]. Available: http://www.honeyd.org/faq.php. [Último acceso: 16 Enero 2021].

[33] P. Niels, «Monkey-Libevent,» 03 Julio 2005. [En línea]. Available: https://monkey.org/~provos/libevent-1.4.14b-stable.tar.gz. [Último acceso: 01 Noviembre 2021].

[34] D. Song, «Source Forge- Libdnet,» 08 Marzo 2000. [En línea]. Available: http://jaist.dl.sourceforge.net/sourceforge/libdnet/libdnet-1.11.tar.gz. [Último acceso: 01 Noviembre 2021].

[35] B. Sources, «Repository- Libpcap,» 04 Agosto 2017. [En línea]. Available: http://repository.timesys.com/buildsources/l/libpcap/libpcap-1.3.0/libpcap-1.3.0.tar.gz. [Último acceso: 01 Noviembre 2021].

[36] N. Provos, «Honeyd Release 1.5c,» 27 Mayo 2007. [En línea]. Available: http://www.honeyd.org/release.php?version=1.5c. [Último acceso: 01 Noviembre 2021].

[37] G. Roelofs, «Zlib,» 15 Enero 2017. [En línea]. Available: https://zlib.net/fossils/zlib-1.2.8.tar.gz. [Último acceso: 01 Noviembre 2021].

[38] R. S. Engelschall, «OpenPKG Project: Arpd,» 07 Abril 2005. [En línea]. Available: http://download.openpkg.org/components/cache/arpd/aprd-0.2.tar.gz. [Último acceso: 01 Noviembre 2021].

[39] P. Niels, «Center for Information Technology Integration-Honeyd,» University of Michigan, 23 Abril 2009. [En línea]. Available: http://www.citi.umich.edu/u/provos/Honeyd/Honeyd_kit-1.0c-a.tgz. [Último acceso: 01 Noviembre 2021].

[40] J. García, «¿Qué es un Script? Todo lo que tienes que saber sobre ellos,» 4 Mayo 2016. [En línea]. Available: https://rootear.com/ubuntu-linux/que-es-un-script. [Último acceso: 5 Noviembre 2021].

ANEXOS

ANEXO 1: CERTIFICADO DE FUNCIONAMIENTO

ESCUELA POLITECNICA NACIONAL
Campus Politécnico "J. Rubén Orellana R

Quito, 23 de diciembre de 2021

CERTIFICADO DE FUNCIONAMIENTO DE PROYECTO DE TITULACIÓN

Yo, Fernando Vinicio Becerra Camacho , docente a tiempo completo de la Escuela Politécnica Nacional y como director de este trabajo de titulación, certifico que he constatado el correcto funcionamiento del módulo virtual en base a los *Honeypots*, a cargo del Estudiante José Jibaja.

El proyecto cumple con los requerimientos de diseño y parámetros necesarios para que los usuarios de la ESFOT puedan utilizar el módulo adecuadamente y alcanzar los objetivos propuestos.

DIRECTOR

Ing. Fernando Vinicio Becerra Camacho., Msc.

Ladrón de Guevara E11-253, Escuela de Formación de Tecnólogos, Oficina 28. EXT: 2729
email: fernando.becerrac@epn.edu.ec Quito-Ecuador