RESUMEN – TRABAJO DE GRADO

AUTORES	**OSCAR JOSE MONTAÑEZ VERJEL** **CARLOS MARIO PAEZ NORIEGA**
FACULTAD	**INGENIERIAS**
PLAN DE ESTUDIOS	**INGENIERÍA DE SISTEMAS**
DIRECTOR	**FABIAN RANULFO CUESTA QUINTERO**
TÍTULO DE LA TESIS	**IMPLEMENTACIÓN DE UNA RED HONEYNET ESTÁTICA APOYADO EN IPV6 PARA ENTORNOS DE RED CABLEADA EN EL LABORATORIO DE REDES Y TELECOMUNICACIONES DE LA UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA.**

RESUMEN
(70 palabras aproximadamente)

ESTE PROYECTO DE INVESTIGACIÓN TIENE COMO OBJETIVO LA REVISIÓN DEL ESTADO DEL ARTE SOBRE LOS ATAQUES INFORMÁTICOS MÁS COMUNES EN REDES CABLEADAS, TIPOS DE HONEYNET ASÍ COMO SU ARQUITECTURA LÓGICA E INFRAESTRUCTURA DE RED, PARA DE ESTA MANERA PROCEDER AL DISEÑO E IMPLEMENTACIÓN DE UNA RED HONEYNET ESTÁTICA BAJO IPV6 PARA ENTORNOS DE RED CABLEADAS ESPECÍFICAMENTE EN EL LABORATORIO DE REDES Y TELECOMUNICACIONES DE LA UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA, CON LA FINALIDAD DE VISUALIZAR Y CONOCER MÁS SOBRE EL ATACANTE, ASÍ PODER TOMAR MEDIDAS DE ASEGURAMIENTO Y PREVENCIÓN PARA GARANTIZAR LA SEGURIDAD E INTEGRIDAD DE LA INFORMACIÓN.

CARACTERÍSTICAS

PÁGINAS: 169	PLANOS:	ILUSTRACIONES:	CD-ROM:1

IMPLEMENTACIÓN DE UNA RED HONEYNET ESTÁTICA APOYADO EN IPV6 PARA ENTORNOS DE RED CABLEADA EN EL LABORATORIO DE REDES Y TELECOMUNICACIONES DE LA UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA.

Autores:

OSCAR JOSE MONTAÑEZ VERJEL

CARLOS MARIO PAEZ NORIEGA

Trabajo de grado presentado para optar el título de Ingeniero de Sistemas

FABIAN RANULFO CUESTA QUINTERO

Ing. De Sistemas

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA

FACULTAD DE INGENIERÍAS

INGENIERÍA DE SISTEMAS

Ocaña, Colombia **Febrero de 2018**

Agradecimientos

A Dios.

Por permitir llegar hasta este punto y habernos dado sabiduría y persistencia necesaria para materializar los conocimiento aprendidos a lo largo de la trayectoria de constante aprendizaje, manteniendo los ánimos ante los momentos de estrés vividos en la realización de la tesis y todo la paciencia que nos fue entregada para no morir en el intento.

A nuestra preciada Universidad Francisco de Paula Santander Ocaña la cual siempre llevamos presente, que nos proporcionó las herramientas y abrió sus puertas del conocimiento para nosotros. Al tan apreciado programa de ingeniería de sistemas de muchos que como nosotros eligieron esta extraordinaria carrera y que con mucho orgullo, amor, pasión y respeto representaremos.

De manera especial dedicamos este triunfo al grupo de investigación GRIITEM por todos los años que hicimos parte de él, en el que fortalecimos y consolidamos habilidades lectoescrituras, destrezas investigativas y disciplinarias en compañía de nuestros docentes y compañeros. Finalmente a los maestros, aquellos que marcaron cada etapa de nuestro camino universitario, y que nos ayudaron en asesorías y dudas presentadas en la elaboración de la tesis. Ms. Fabián Cuesta Quintero quiero por su gran apoyo y motivación para la culminación de nuestros estudios profesionales y para la elaboración de esta tesis; Ph. Dewar Wilmer Rico Bautista por su tiempo compartido y por impulsar el desarrollo de nuestra formación profesional; Esp. Luis Anderson Coronel Rojas por apoyarnos en su momento; Ms. Edwin Barrientos Avendaño por su apoyo ofrecido en este trabajo de grado.

A nuestros padres por ser el pilar fundamental en todo lo que somos, en toda nuestra educación, tanto académica, como de la vida, por su incondicional apoyo perfectamente mantenido a través del tiempo.

Todo este trabajo ha sido posible gracias a ellos.

Índice

Lista de Tablas

Lista de Figuras

Resumen

Este proyecto de investigación tiene como objetivo la revisión del estado del arte sobre los ataques informáticos más comunes en redes cableadas, tipos de Honeynet así como su arquitectura lógica e infraestructura de red, para de esta manera proceder al diseño e implementación de una red Honeynet estática bajo IPv6 para entornos de red cableadas específicamente en el laboratorio de redes y telecomunicaciones de la Universidad Francisco de Paula Santander Ocaña, con la finalidad de visualizar y conocer más sobre el atacante, así poder tomar medidas de aseguramiento y prevención para garantizar la seguridad e integridad de la información.

PALABRAS CLAVE

Honeynet, Honeypot, Honeywall, IPv6

Introducción

Muchas empresas alrededor del mundo han tenido que lidiar con los ataques informáticos debido a la constante existencia de estos en sus sistemas de información y que a su vez sus datos se ven comprometidos debido a las vulnerabilidades y los agujeros de seguridad que existen en sus sistemas de redes que basan su funcionamiento a través del estándar de comunicaciones TCP/IP. La seguridad informática es un aspecto fundamental a considerar y un tema de gran enfoque y preocupación en las distintas empresas que existen en el mundo porque los métodos, herramientas y/o procedimientos a realizar son los que van a defender y proteger su información.

El nuevo protocolo de internet IPv6, fue diseñado con la finalidad de mitigar los fallos que se presentan en la estructura del anterior protocolo IPv4, para garantizar que el uso de las tecnologías de la información y la comunicación en las instituciones sea un elemento a favor del buen funcionamiento de los procesos internos y externos. Entonces es necesario implementar herramientas bajo IPv6 que permitan identificar las vulnerabilidades y las estrategias que utilizan los atacantes para penetrar los sistemas.

"Cuando un oficial de seguridad quiere asegurar una organización, debe ser consciente de todas las amenazas potenciales, incluso si esta amenaza es un protocolo de diez años que representa menos del 1 por ciento del tráfico total de Internet en 2008. No se ciegue por este 1 por ciento: Esta cifra está condenada a aumentar en los próximos años, y es probable que su red ya está expuesta a algunas amenazas IPv6. Es mejor estar seguro que lamentarlo." (Hogg, IPv6 Security, 2008)

Capítulo 1. Implementación de una red HONEYNET estática apoyado en IPV6 para entornos de red cableada en el laboratorio de redes y telecomunicaciones de La Universidad Francisco de Paula Santander Ocaña

1.1 Planteamiento Del Problema

Los datos que viajan a través de internet se exponen a múltiples riesgos debido a la vulnerabilidad de las redes, pues existen muchas personas que se dedican a robar y destruir todo tipo de información. Estos pueden ser gente de la misma compañía o empresa que tienen fácil acceso a la red y también existen personas externas que logran violar la seguridad de la red (Argueta, 2001). Según el manual de seguridad en redes, de la Coordinación de Emergencia en Redes Teleinformáticas de la Administración Pública Argentina, afirma que la falta de medidas de seguridad en las redes es un problema que está en crecimiento. Pues cada vez es mayor el número de atacantes y cada vez están más organizados, por lo que van adquiriendo día a día habilidades más especializadas que les permiten obtener mayores beneficios. Tampoco deben subestimarse las fallas de seguridad provenientes del interior mismo de la organización (ARCERT, 2008). (Gáldamez, 2013) expresa que "Los efectos de las diversas amenazas pueden ser muy variados. Unos pueden comprometer la integridad de la información o de los sistemas, otros pueden degradar la disponibilidad de los servicios y otros pueden estar relacionados con la confidencialidad de la información". (p. 2)

Hoy en día, la información está en el corazón de todas las economías. Las sociedades modernas deben mantener el ritmo con el crecimiento del conocimiento. El laboratorio de redes y telecomunicaciones de la Universidad Francisco de Universidad Francisco de Paula Santander Ocaña, no es la excepción. Institución educativa que gestiona una gran cantidad de información

proveniente de funcionarios, docentes, estudiantes, transacciones y demás procesos que generen algún tipo de información que requiera ser almacenada y gestionada, esta es expuesta a una innumerable cantidad de ataques que ponen en riesgo la confidencialidad, la disponibilidad o la integridad de los datos. “Mientras que Internet y las tecnologías digitales facilitan el acceso al conocimiento, al mismo tiempo hay ciertas barreras que impiden el acceso”. (Miguel, Gomez, & Bongiovani, 2012, pág. 2)

Con el surgimiento de nuevos protocolos de comunicaciones como lo es IPv6 que de la misma manera promete el aumento de usuarios conectados a la red, también aumentará el nivel amenazas para los cibernautas y la información que en ella circula. Hogg & Vyncke exponen, “IPv6 es el segundo protocolo estándar de capa de red que sigue a IPv4 para comunicaciones de computadoras a través de Internet y otras redes informáticas.”, por lo cual estar al tanto de las vulnerabilidades, posibles ataques, métodos y técnicas podrá colocar a la institución un paso adelante de los atacantes para garantizar la confianza de sus usuarios. (Hogg & Vyncke, IPv6 Security, 2008, pág. 3)

El Ministerio de Tecnologías de la Información y las Comunicaciones de Colombia, busca promover la adopción de IPv6 en las entidades e instituciones educativas de todo el país, para realizar esta migración de IPv4 a IPv6 se debe seguir las instrucciones citadas en la circular 002 del 6 de julio del 2011, El protocolo IPv6 proporciona unos niveles de seguridad mucho más altos que su antecesor, pero a su vez existe desconocimiento sobre los ataques realizados bajo este protocolo. Siendo evidente la necesidad de Universidad Francisco de Paula Santander Ocaña, migrar al protocolo de comunicaciones versión 6 (IPv6), afrontando retos como lo es el desconocimiento en cuanto a las posibles vulnerabilidades. Cuando se trabaja bajo ambientes de IPv4 se cuenta con un amplio conocimiento de defensa para los ataques debido a sus vulnerabilidades, que son aprovechadas por los hackers que al implementar métodos y técnicas,

acceden a la red e información. Por lo contrario, IPv6 por ser nuevo en su implementación, se encuentra en evolución así que no se conoce la totalidad de sus vulnerabilidades. Al implementar IPv6 sin tener un conocimiento sólido de los riesgos a los que se expone la información almacenada en los servidores se expone a situaciones críticas en las que el no saber qué hacer en casos en las que un atacante ha logrado penetrar el sistema de información.

En tal sentido, se hace imperativo proponer un mecanismo que permita revisar eventos de seguridad que conlleven a identificar usos indebidos o fuera de lo normal para determinar las estrategias usadas por los atacantes para penetrar un sistema. Para ello, surgen las redes señuelo que son herramientas de seguridad útiles que permiten monitorear actividades maliciosas para reconocer el patrón de un ataque y descubrir los nuevos métodos. Además al ser una institución educativa, la Honeynet ofrecerá a los estudiantes de carreras tecnológicas una herramienta útil para evaluar en un ambiente real los problemas de seguridad.

1.2 Formulación Del Problema

¿Se podrá implementar una red Honeynet que permita el manejo del protocolo IPV6?

1.3 Objetivos

1.3.1 Objetivo General

Diseñar e implementar una red Honeynet estática para entornos de red cableada apoyado en IPv6, en el laboratorio de redes y telecomunicaciones de la universidad francisco de paula Santander Ocaña.

1.3.2 Objetivos Específicos

Caracterizar vulnerabilidades y ataques más comunes en redes cableadas IPv6 que permitan identificar patrones en procedimientos de ataques.

Implementar un modelo de red Honeynet estática diseñado para el laboratorio de redes y telecomunicaciones de la Universidad Francisco de Paula Santander Ocaña.

Evaluar el funcionamiento de la red Honeynet, analizando métodos y técnicas de atacantes obtenidos a través de esta, para plantear medidas de aseguramiento.

1.4 Justificación

“La seguridad absoluta, tendría un costo infinito.” Anónimo.

El sector de las telecomunicaciones es algo indispensable en la vida actual, a raíz de la introducción de la informática en los hogares y los avances tecnológicos, ha surgido toda una generación de personajes que difunden el miedo en la Red y/o cualquier sistema de cómputo. Todos ellos son catalogados como "piratas informáticos" o “piratas de la Red” la nueva generación de "rebeldes" de la tecnología aportan, unos sabiduría y enseñanza que difunden, otros destrucción o delitos informáticos. Hay que saber bien quién es cada uno de ellos y catalogarlos según sus actos de rebeldía en la mayoría de los casos. (Arango, 2010, pág. 2)

Un Honeypot es un tipo de mecanismo informático que sirve para detectar o frustrar intentos no autorizados de acceso a sistemas informáticos. Actúa como un tarro de miel o anzuelo provocativo para atraer al enemigo (a diferencia de los sistemas de defensa comunes como los Firewall) y aprender de él, pues esto supone una parte más humana. Visto así, Honeynet es el sistema de red que se compone de varios Honeypots.

Para justificar este proyecto hay que pensar en la importancia teórica de la implementación de Honeynet en futuras investigaciones, puesto que uno de los objetivos es obtener información, conocer el comportamiento de los atacantes y plantear estrategias para proteger la información en el campo de IPv6.

En la actualidad la universidad cuenta con un Backbone en fibra óptica y un cableado horizontal en categoría 7e que permite anchos de banda eficaces y que muy pocas universidades poseen, lo que indica que se cuenta con la infraestructura apropiada para soportar entornos IPv6, pero por otro lado, el ya agotado protocolo IPv4 es aún utilizado en la universidad y no se cuenta con una preparación para migrar a IPv6. En IPv4 al momento de ocurrir ataques ya se sabe qué hacer para proteger la información, pero por el contrario bajo un ataque IPv6, no. Así que es necesario actualizarse a este protocolo para lograr mejoras en la seguridad informática y estimular la investigación en este campo.

Honeynet opera bajo IPv6, por consiguiente sería un sistema ideal para solucionar esta problemática. Además como se mencionaba antes, también se busca aprender del atacante y conocer su modus operandi. Hay mecanismos comunes de defensa como los Firewall en IPv4, los cuales simplemente bloquean todo y mantienen alejado al atacante todo el tiempo haciendo imposible conocer sus tácticas y cuando este es capaz de penetrar, el Firewall se vuelve inservible. Se pueden realizar pruebas controladas de ataques, por ejemplo pedirle a la unidad de División de Sistemas de la universidad que realice un ataque a la red Honeynet para verificar su funcionamiento sin necesidad de esperar a que ocurra un ataque real.

Para concluir, esta propuesta de carácter investigativo mejora varios aspectos, entre los cuales se incluye: Conveniencia: al mejorar la seguridad de las redes de computadores mediante el uso de Honeynet. Y el aspecto social: debido a que se genera confianza, seguridad y protección

de la información sobre todo si es de carácter sensible, esto también incluye mejora de calidad de procesos y transmisión de datos.

1.5 Hipótesis

Si se implementan el protocolo IPV6 en la red Honeynet, se podrá observar el mejoramiento de la seguridad de los sistemas.

Al implementar una red Honeynet se resolverá el problema en caso de ataques realizados bajo IPV6.

Honeynet implica un avance de seguridad significativo en infraestructura.

Al aplicar la red Honeynet, humanamente se aprenderá mucho del comportamiento de los atacantes y también esta información resultara útil en el ámbito investigativo.

1.6 Delimitación

1.6.1 Operativa.

Esta propuesta está limitada a la implementación y evaluación de una Honeynet estática apoyada en el protocolo IPv6 para entornos de red cableados en el laboratorio de redes y telecomunicaciones de la Universidad Francisco de Paula Santander Ocaña.

1.6.2 Conceptual.

Los temas a tratar en esta propuesta exploran el campo de la seguridad en redes de computadores, sus respectivos protocolos como el de red TCP/IP y de internet IPv6, siendo éstos aspectos fundamentales en la seguridad informática y el énfasis de este proyecto bajo la perspectiva de Honeynet estática.

1.6.3 Geográfica.

Todos los procedimientos, métodos y herramientas de seguridad basados en Honeynet estática, se implementarán en laboratorio de redes y telecomunicaciones de la Universidad Francisco de Paula Santander Ocaña.

1.6.4 Temporal.

La investigación tendrá un tiempo de realización de 5 meses, de acuerdo con las diferentes actividades a realizar durante el desarrollo de la misma. A partir de la aprobación del proyecto.

Capítulo 2. Marco Referencial

2.1 Marco Histórico

2.1.1 Problemática de Ataques

En la actual era digital, como secuelas del uso ineludible de las tecnologías de la información y las comunicaciones, se desconoce la totalidad de sus consecuencias. Sin embargo esto no ha sido un limitante en su evolución y utilización en innumerables procesos llevados a cabo alrededor del mundo. Debido al gran volumen de datos generados diariamente en la sociedad, se implementan sistemas informáticos encargados de gestionar información y optimizar procesos. En consecuencia, no es de extrañar que la gestión de seguridad de la información, donde el propósito es proteger los activos de información de una organización, se ha convertido en una cuestión estratégica importante para la mayoría de las organizaciones. (Von solms, 2013).

El avance vertiginoso de la tecnología; así mismo como el famoso Internet de las Cosas (IoT), también ofrece oportunidades inmensas a los criminales informáticos, ya que todo se encuentra cada vez más digitalizado y por consiguiente hay más vulnerabilidad de acceso a la información siempre que se logre el propósito de alcanzar un nuevo mecanismo que supere, mejore o refuerce a su versión anterior. También se suma a esto que es imposible obtener un cien por ciento (100%) de seguridad informática para la protección de la información confidencial, siempre se logran altos índices de seguridad, pero por más alto que sea este porcentaje nunca va a alcanzar esa totalidad de seguridad, se generan así “agujeros de seguridad” el cual es un término general para referirse a lo que son los pequeños puntos débiles por donde un atacante fijara su atención y por medio de este, puede ejercer sus funciones criminales a un determinado sistema informático.

Es allí cuando es conveniente ponerse en los zapatos de este criminal y pensar en cómo va actuar y qué posibles acciones tomará para penetrar el sistema informático por medio de un agujero de seguridad; es necesario un monitoreo en el atacante. De esta forma se podría analizar sus conductas y predecir algunas de sus acciones, y qué mejor forma de vigilarlo que atraerlo directamente hacia su objetivo para estudiar sus movimientos y así poder detenerlo y denunciarlo a las autoridades; aprender de él y atrapar futuros invasores.

¿No sería mejor conocer al criminal trayéndolo directamente a su carnada?

2.1.2 The Honeynet Project

The Honeynet Project es una organización internacional de investigación de seguridad sin fines de lucro dedicada a investigar los últimos ataques y desarrollar herramientas de seguridad de código abierto para mejorar la seguridad en Internet. Con Capítulos de todo el mundo, sus voluntarios han contribuido a luchar contra el malware (como Confickr), descubrir nuevos ataques y crear herramientas de seguridad utilizadas por empresas y agencias gubernamentales de todo el mundo. La organización sigue estando a la vanguardia de la investigación de seguridad, trabajando para analizar los últimos ataques y educar al público sobre las amenazas a los sistemas de información en todo el mundo.

Fundada en 1999, The Honeynet Project ha contribuido a luchar contra el malware y los ataques maliciosos de la incursión y tiene el profesional principal de la seguridad entre los miembros y los alumnos. Su misión es "aprender de las herramientas, las tácticas y los motivos involucrados en ataques informáticos y de red, además de compartir las lecciones aprendidas" con tres pilares principales: Investigación, conciencia y suministro de herramientas.

Su visión se define como: “The Honeynet Project es un grupo diverso, talentoso y comprometido de expertos internacionales en seguridad informática que llevan a cabo una

investigación y desarrollo abiertos y transversales en el panorama de la amenaza en evolución. Coopera con personas y organizaciones de ideas afines en ese empeño.” (The honeynet proyect, 1999)

“The Honeynet Project” fue iniciada para encender una luz en esta oscuridad. Este equipo de investigadores ha construido una red informática completa y totalmente cableada con sensores. Luego pusieron la red en Internet, dándole un nombre y un contenido adecuadamente atractivos, y registró lo que pasó. (La dirección IP actual no se publica y cambia con regularidad.) Las acciones de los hackers se registran a medida que suceden: cómo intentan entrar, cuando tienen éxito, y qué hacen luego de obtener el éxito. (Scheiner, 2001)

2.1.3 Evolución del Internet de las Cosas (IoT)

En el 2008 el número de cosas conectadas a la red superó el número de personas que habitaban la tierra, esto supuso también el agotamiento paulatino de las direcciones reales IPv4 y por ende el paso a la siguiente generación de sistemas en entornos IPv6. En una casa pueden haber varios objetos como computadores, televisores, consolas, celulares, tablets, relojes y electrodomésticos, todos estos en la actual era tecnológica se conectan a internet lo cual en los tiempos que IPv4 fue planteado era inimaginable, también el uso de tecnología inalámbrica y el uso de protocolos de este tipo como WI-FI y Bluetooth supusieron ese avance vertiginoso y potencial de la red. Conceptos como M2M (Machine To Machine) y Web Semántica fueron eventualmente reemplazados por el de Internet de las Cosas (IoT por sus siglas en inglés Internet of Things) y redefinidos como parte de este.

Aun así el 99% de cosas físicas en este mundo no están conectadas a internet, eventualmente

a lo largo de la historia de la humanidad, CISCO prevé que este porcentaje

disminuya y se llegue así a un nuevo nivel, a un nuevo mundo conocido anticipadamente como IoE (Internet del Todo, Internet of Everything en inglés), pero mientras esto sucede es necesario adaptar nuevos estándares de seguridad preferiblemente en IPv6, ya que sería una utopía llegar a tal avance tecnológico sin que existan atacantes y criminales electrónicos, gente que utiliza su inteligencia y estudio para el mundo del mal llegaran a medida que se logre un nuevo objetivo o paso en el mundo digital.

"En analogía con la definición de universo, el cual se define comúnmente como la totalidad de la existencia, un universo de Internet de las Cosas podría conectar potencialmente todo. Como una analogía adicional a nuevas teorías sobre universos paralelos, diferentes mundos de Internet de las Cosas pueden desarrollarse y existir en paralelo, potencialmente se superponen y poseen puertas de transferencia espontáneas o fijas". (Vermesan & Friess, 2013)

"Básicamente, IPv6 puede hacernos más seguros, pero solo si lo hacemos bien. Los ataques web no desaparecerán y los NULL (o cifrados de cifrado de 0 bits) aún pueden ser implementados por administradores no observadores, pero hay un camino a seguir si comenzamos a planear ahora." (Wright, 2011)

2.2 Marco Contextual

El Laboratorio de radio y telecomunicaciones (LRYT) ubicado en la Universidad Francisco de Paula Santander Ocaña, cuenta con dispositivos físicos como Switch y Router de la marca CISCO y un servidor de la marca Hewlett Packard las cuales son empresas reconocidas en el mercado y de alta calidad para las redes de información, dicho servidor tiene instalados los sistemas operativos Windows Server 2000 y Ubuntu 16.4 LT Donde este último será utilizado para aplicar la red Honeynet.

Se resalta que el Laboratorio de radio y telecomunicaciones está disponible para todos los estudiantes pertenecientes a las carreras de ingeniería de sistemas y técnicos en telecomunicaciones y dicho proyecto beneficiará a tal comunidad.

2.3 Marco Conceptual

2.3.1 Seguridad de la Información

"La Seguridad de la información es el conjunto de estándares, procedimientos, estrategias, recursos informáticos, educativos y humanos integrados para proveer toda la protección debida y requerida a la información de una empresa, institución o agencia gubernamental". (Rodriguez, 2011)

2.3.2 Intrusos Informáticos

Los intrusos informáticos son las persona que intentan acceder a un sistema sin contar con ningún tipo de autorización, también se les clasifica como criminales y atacantes.

2.3.3 Definición de Honeypot

"Se denomina Honeypot (tarro de miel) al recurso de red destinado a ser atacado o comprometido con la finalidad de identificar, evitar y en cierta medida, neutralizar los intentos de secuestrar sistemas y redes de información". (Edgar A, 2012)

2.3.4 Definición de Honeynet

Una Honeynet básicamente es una red de Honeypots que tiene como fin el proporcionar información valiosa sobre los métodos y recursos utilizados por la comunidad Blackhat para cometer ataques informáticos. Se las conoce también como Honeypots de alta interacción. Reflejan un entorno de red productivo al trabajar con varios sistemas a la vez. Entre ellos Linux, Solaris, Windows, Router Cisco, etc. (Edgar A, 2012)

2.3.5 Definición de Honeywall

El Honeywall es el principal componente de la arquitectura; actúa como puente transparente de la Honeynet y ejecuta las tareas de control, captura y análisis de los datos. Se implementa utilizando el sistema operativo Honeywall Roo V1.4 basado en CentOS 5.0 distribuido de forma gratuita por el proyecto Honeynet "The Honeynet Project". (Edgar A, 2012)

2.4 Marco Teórico

2.4.1 Honeypot

Un Honeypot es un tipo de mecanismo informático que sirve para detectar o frustrar intentos no autorizados de acceso a sistemas informáticos, estos tipos de acceso también son conocidos como ataques informáticos. Este mecanismo consiste en datos de información que se encuentran alojados en lugares específicos tales como servidores o páginas web de forma normal como cualquier otro tipo de dato que se encuentre allí, pero el Honeypot es secretamente aislado y monitoreado para que capture información de posibles atacantes cuando estos intenten acceder a al Honeypot y así enviar en detalle esta información al administrador de la red.

De allí la etimología de su nombre en inglés la cual se traduce al español como "tarro de miel", que en el contexto informático es un anzuelo que atrae atacantes y permite adquirir información de gran valor de este para luego poder atraparlo, bloquearlo y reportarlo a autoridades sin que este se dé cuenta que está siendo vigilado, pues el robo de información es considerado un crimen.

Resulta curioso que la mayoría de herramientas que se utilizan en el mundo de la seguridad informática funcionan bajo la premisa de mantener alejados a los atacantes, mientras que el

concepto de Honeypot que a su vez es una de estas herramientas contradice esta premisa pues hace todo lo contrario de atraer un atacante y analizarlo desde cerca sin que este se percate.

2.4.1.1 Clasificación de Honeypots

Debido a que los Honeypots pueden ser simples o complejos; se clasifican de las siguientes maneras:

2.4.1.1.1 Clasificación A: Según su Interacción

I) Honeypots de Alta Interacción: Este tipo de Honeypots están diseñados exclusivamente para que sean atacados y configurados para que cuando esto suceda envíen alertas al más mínimo intento de acceso a ellos, generalmente son máquinas reales (físicas) usadas por empresas en sus arquitecturas de redes internas. Aunque puede verse como un arma de doble filo, la gran ventaja de este tipo de Honeypots es que pueden prevenir cualquier tipo de ataques. En los Honeypots de alta interacción todas las fallas están al descubierto así que si no se hace el correcto aislamiento y monitoreo de este, toda la información se podría ver comprometida y el culpable de esto no solo sería el atacante sino además el administrador.

II) Honeypots de Baja Interacción: Estos Honeypots suelen ser creados y gestionados por organizaciones dedicadas a la investigación del fraude en Internet, o cualquier tipo de organización que necesite investigar sobre las nuevas amenazas en la red. Esto es contraparte de los Honeypots de Alta Interacción debido a su complejidad y su recolección de información profunda y detallada. Generalmente se tratan de sistemas que emulan servicios, protocolos etc… Pero siempre existe un proceso en un segundo plano el cual se le denomina como un "meta-sistema", puesto que ese sistema o ese conjunto de sistemas son Honeypots emulan y simulan a la vez la función de dicho protocolo o servicio, Por ejemplo si un Honeypot de Baja Interacción está

como un protocolo de mensajería este recibe su información del usuario pero no la envía. Dejando como conclusión que esta clase de Honeypots se dedica es a capturar herramientas automatizadas y no personas en sí; cada vez se vuelve más complicado detectar nuevos ataques por lo que la clave es detectar patrones de ataques. Debe estar altamente protegido para que no se ponga en contra del administrador.

Alta interacción	Baja interacción
Servicios reales, sistemas operativos o aplicaciones	Emulan servicios, vulnerabilidades, etc.
El riesgo que corren es mayor	El riesgo que corren es menor
Capturan menos información, pero más valiosa	Capturan mucha información. Dependen de su sistema de clasificación y análisis para evaluarlo.

Fuente: Virtual Honeypots

***Figura 1*:** Comparación entre Honeypots de alta y baja interacción, Fuente: **Virtual Honeypots**

III) Honeypots Puros: Son sistemas de producción completos. Las actividades del atacante son monitoreadas usando un toque casual que ha sido instalado en el enlace del Honeypot a la red. No es necesario instalar ningún otro software. Aunque un Honeypot puro es útil, la furtividad de los mecanismos de la defensa puede ser asegurada por un mecanismo más controlado.

IV) Honeymonkey: Aunque no es un Honeypot en sí, está basado en la emulación de Servidores. En este entorno se añade el término Honeypot Cliente, puesto que cambia de bando y no espera a ser atacado si no a atacar y estar bastante activo en esta práctica, esto es un Honeymonkey. Tiene los mismos objetivos que los Honeypots mencionados, solo que varía en su función, ya que la exploración se hace activamente por medio del navegador, esto es que siempre está visitando todo tipo de páginas y si encuentra una que detecta vulnerabilidades de tal navegador hace que esta intente aprovechar esa oportunidad y así recolectar información. Son

demasiado activos por los que se le comparan como agentes que siempre están vigilando y patrullando un sistema de red. Fue Microsoft quien los bautizó. “Monkey” (mono, en inglés) hace alusión a los saltos y el dinamismo del tipo de acción que realizan. Con este método, al igual que los Honeypots, se pueden encontrar nuevos Exploits, gusanos, etc., siempre que se analice y procese convenientemente toda la información recogida. (HONEYPOTS, MONITORIZANDO A LOS ATACANTES, 2012)

V) Honeytoken: Es un término teórico para referirse a los Honeypots que no son computadores, representados principalmente por bases de datos, nubes de almacenamiento o información que está siendo accedida abusivamente, un Honeytoken es un testigo pasivo de ataques; no los previene pero si da ideas a los administradores y un mejor dimensionamiento confidencial de la integridad de datos.

2.4.1.1.2 Clasificación B: Según su Valor de Seguridad

I) Honeypots de Producción: Son aquellos que se encarga de capturar y defender, proporcionan servicios que están presentes en una red autentica, estos Honeypots cuentan con varias tareas, una de ellas es reducir el riesgo de ataques en la red principal de la empresa, previenen el acceso a información confidencial mediante el engaño a los atacantes y la prevención adecuada. Hacen que los atacantes crean que están cumpliendo su objetivo de atacar un sistema y robar su información cuando en realidad están es atacando un señuelo. Cuando esto sucede se utilizan métodos como la denegación y limitación de servicios o incluso la prohibición temporal de estos, para no solo para evitar el acceso no autorizado si no recopilar información de todos los detalles de las opciones y herramientas utilizadas por el atacante.

II) Honeypots de Investigación: como su nombre lo indica son Honeypots utilizados por las instituciones con énfasis investigativo y científico para proteger los sistemas de nuevos ataques y

amenazas. Su función y enfoque principal es analizar la información obtenida, también puede generar un historial paso a paso de los movimientos realizados por el criminal para así elaborar un perfil que concuerde con su modus operandi.

2.4.1.1.3 Clasificación C: Según su Estado

I) Honeypots físicos: Aquellos que son implementados en una máquina real, este a su vez tiene una interacción bastante alta y pueden ser comprometidos en su totalidad. Su instalación es cara y más elaborada.

II) Honeypots virtuales: Son aquellos que se implementan por medio de la asignación de direcciones IP debido a la amplia cantidad de espacio y disponibilidad de estas direcciones en entornos virtuales.

Cabe resaltar que en una máquina física se pueden emular Honeypots virtuales por medio de máquina virtuales, al realizar esto, el Honeypot virtual adquiere casi en su totalidad las propiedades y ventajas concedidas por un Honeypot físico real de alta interacción y de forma más económica-practica utilizando menos hardware. Así como en una máquina virtual, estos Honeypots emulados dependen de un software pero también se les da conexión a la red y su propio rango de direcciones IP.

Debido a que no hay razones legítimas para conectarse a un Honeypot, cualquier interacción es probablemente maliciosa. Por lo tanto, los Honeypots reducen dramáticamente el número de alertas falsos positivos en comparación con los productos de seguridad basados en eventos de red tradicionales, como los sistemas de detección de intrusos. Esta alta relación señal-ruido es útil para los administradores a menudo abrumados por falsas alarmas, ya que los

conjuntos de datos más pequeños son relativamente pequeños y fáciles de manejar y analizar. Esto da como resultado una detección de ataque más corta y tiempos de respuesta a incidentes. (David Watson, 2008)

2.4.1.2 Arquitectura de un Honeypot

Hay que tener en cuenta que los Honeypots se deben integrar con el resto del sistema que se tiene implementado, por ejemplo: servidores web, servidores de ficheros, DNS. De manera de asegurar que no interfiera con las otras medidas de seguridad que puedan ya existir en la red como Firewalls, IDS. (Mora, 2009)

Los Honeypots pueden servir tanto para la detección de atacantes internos como externos. Se debe tener siempre en cuenta la posibilidad de establecer Honeypots internos para la detección de atacantes o sistemas comprometidos en la red, por ejemplo sistemas infectados con gusanos o virus. (Mora, 2009)

Un Honeypot puede ubicarse antes del firewall, después del firewall y en la zona desmilitarizada. La primera localización (***Figura 2***) permitirá evitar el incremento del riesgo inherente a la instalación del Honeypot. Como este se encuentra fuera de la zona protegida por el firewall, puede ser atacado sin ningún tipo de peligro para el resto de la red. Esta configuración evitará las alarmas de otros sistemas de seguridad de la red (IDS) al recibir ataques en el Honeypot. Sin embargo, existe el peligro de generar mucho tráfico debido precisamente a la facilidad que ofrece el Honeypot para ser atacado. (Mora, 2009)

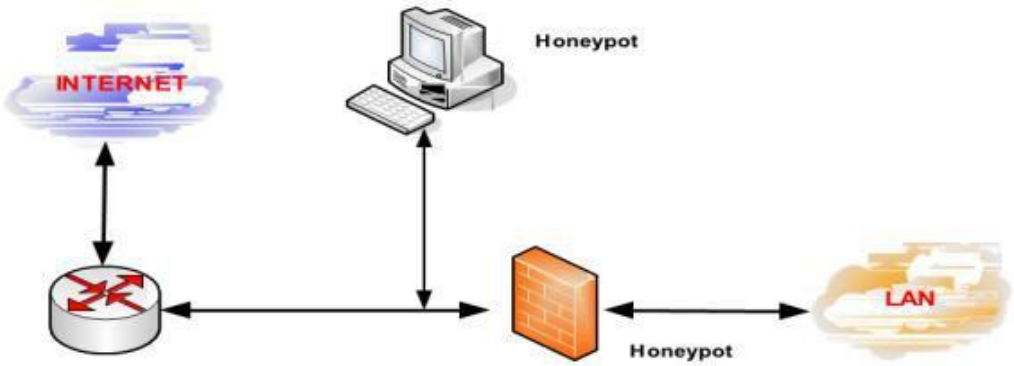

Figura 2. Ubicación del Honeypot antes del firewall, Fuente: **P. Mora, Honeypots Seguridades Informáticas**

Detrás del firewall (***Figura 3***) el Honeypot queda afectado por las reglas de filtrado del firewall. Por un lado se tiene que modificar las reglas para permitir algún tipo de acceso al Honeypot por posibles atacantes externos, y por el otro lado, al introducir un elemento potencialmente peligroso dentro de la red se puede permitir a un atacante que gane acceso al Honeypot y a la red. La ubicación tras el firewall permite la detección de atacantes internos así como firewalls mal configurados, máquinas infectadas por gusanos o virus e incluso atacantes externos. (Mora, 2009)

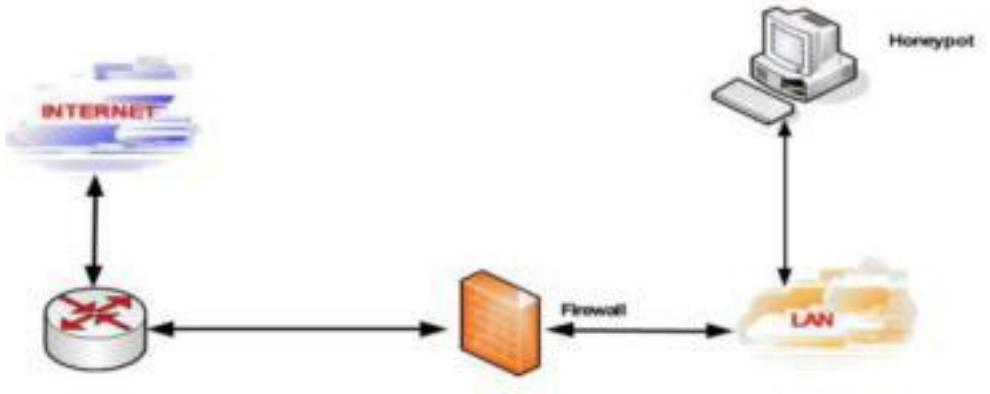

Figura 3. Ubicación del Honeypot después del firewall, Fuente: **P. Mora, Honeypots Seguridades Informáticas**

La última ubicación (***Figura 4***) permite por un lado juntar en el mismo segmento a los servidores de producción con el Honeypot, y por el otro, controlar el peligro que añade su uso, ya que tiene un firewall que lo aísla del resto de la red local. Esta arquitectura permite tener la posibilidad de detectar ataques externos e internos con una simple reconfiguración del sistema de firewall puesto que se encuentra en la zona de acceso público. (Mora, 2009)

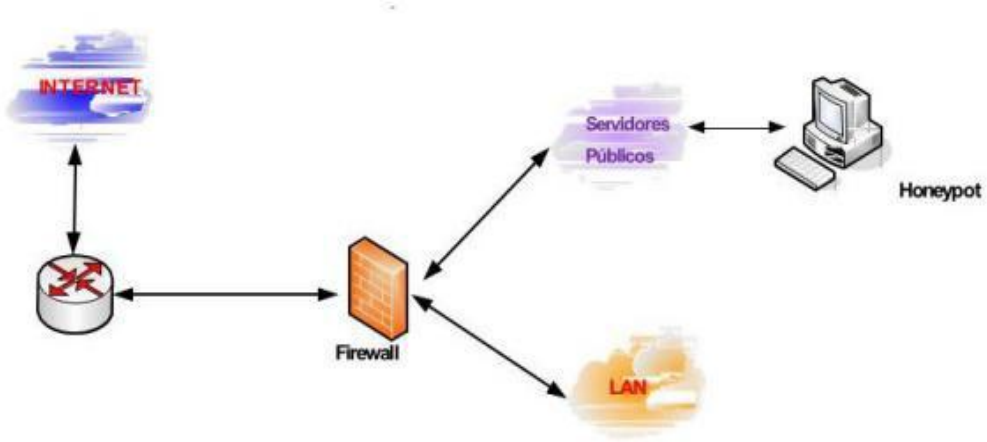

Figura 4. Ubicación del Honeypot en la zona desmilitarizada, Fuente: **P. Mora, Honeypots Seguridades Informáticas**

2.4.2 IPv6

El Protocolo de Internet (IP) es el protocolo de comunicaciones más utilizado. Debido a que es la tecnología de comunicación más ubicua, es el enfoque de cientos de miles de profesionales de TI. Debido a que mucha gente confía en el protocolo, la seguridad de las comunicaciones está en la cima de lo que se debe pensar. La investigación de seguridad que se lleva a cabo en el protocolo IP es conducida por personas benévolas y malévolas. Toda la investigación de seguridad ha causado muchos parches y ajustes a IP, ya que se ha desplegado internacionalmente. En retrospectiva, habría sido mejor si una consideración más profunda se dio a la seguridad del protocolo antes de que fuera ampliamente desplegado. (Hogg, IPv6 Security, 2008)

El Protocolo de Internet versión 6 (IPv6) es la nueva actualización del protocolo IP en su versión 4 (IPv4), está definida por el documento RFC 2460, el cual resuelve principalmente el agotamiento de direcciones asignadas por el IPv4, cuando se pensaba en un principio que nunca se iban a agotar. IPv4 posibilita 4 294 967 296 (232) direcciones de host diferentes, un número inadecuado para dar una dirección a cada persona del planeta, y mucho menos a cada dispositivo como teléfonos, relojes, tablets, etcétera.

Por otro lado, IPv6 admite 340.282.366.920.938.463.463.374.607.431.768.211.456 (2128 o 340 sextillones de direcciones) cerca de 6,7 × 1017 (670 mil billones) de direcciones por cada milímetro cuadrado de la superficie de la Tierra.

2.4.3 Honeynet

Una Honeynet es un tipo concreto de Honeypot. Específicamente, es un Honeypot altamente interactivo diseñado para la investigación y la obtención de información sobre

atacantes. Una Honeynet es una arquitectura y no un producto concreto o un software. El nuevo enfoque no consiste en poner datos falsos o engañar a un posible atacante (como suelen hacer algunos Honeypot) sino que el objetivo principal es recolectar información real de cómo actúan los atacantes en un entorno de verdad. Para conseguir este entorno real con sistemas reales, no con simples emulaciones de servicios y altamente interactivo, se dispone una configuración de red típica con todos los elementos. (Cybsec Security Sustems, 2013)

Una red Honeynet está diseñada para que sea vulnerable y su información esté comprometida, por consiguiente debe estar aislada y monitoreada. Pero esto indica que si se deja así tal y como está el atacante va a sospechar que está siendo vigilado, así que es necesario añadir los otros elementos que conforman una arquitectura de red normal para que dicho atacante crea estar ante una topología real a la cual puede acceder.

2.4.3.1 Arquitectura Honeynet

Las Honeynets no son un producto, son toda una arquitectura, una red con un ambiente totalmente controlado, dentro de ella tenemos a los sistemas que son los objetivos. La Honeynet es como una pecera con servidores, Router, computadores personales, y todos los elementos de una red común dentro de ella como elementos, mientras nosotros vemos cómo los atacantes interactúan con ellos. (Spitzner, 2003)

Para mantener el ambiente controlado la clave en la arquitectura de la Honeynet es su puerta de enlace llamado Honeywall. Este dispositivo separa la Honeynet del resto del mundo. El Honeywall es un dispositivo que originalmente era de Capa 3 pero actualmente puede ser un bridge invisible de Capa 2. Tiene tres interfaces de red (eth0, eth1, eth2) como se muestra en la ***Figura 5***; las dos primeras (eth0, eth1), como puertas de entrada y salida, forman el bridge de

Capa 2 separando la Honeynet con el mundo, y una tercera interfaz de red opcional que sirve para administración.

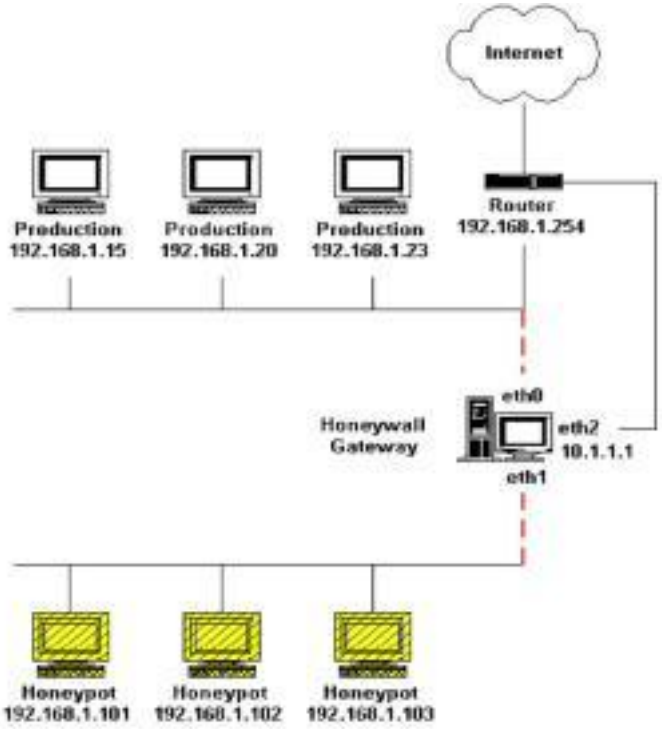

Figura 5. Arquitectura general de una Honeynet, Fuente: **The Honeynet Project**

Para garantizar el correcto funcionamiento de la Honeynet y mantener un ambiente seguro para los sistemas contiguos a la red. Estos requisitos son: control de datos, captura de datos y recolección de datos.

2.4.3.1.1 Honeynet de Generación I (GEN I)

Fue desarrollada por The Honeynet Project en el año 1999. Este tipo de arquitectura incorpora de una forma sencilla el control y la captura de datos, permitiendo la recopilación máxima de las actividades efectuadas por los atacantes y simulando un ambiente real. En la ***Figura 6*** se muestra una Honeynet de Primera Generación que requiere dos interfaces de red en

su puerta de enlace, una que se muestra hacia la red externa, y la otra que lo hace hacia la red interna, constituida por varios Honeypots. Las actividades de control y captura de datos las realiza un Firewall de capa tres, que actúa a su vez como una puerta de enlace en modo de Traductor de Direcciones de Red (NAT, Network Address Translation). Como desventaja de esta arquitectura está el hecho de que puede ser detectada por intrusos con conocimientos avanzados.

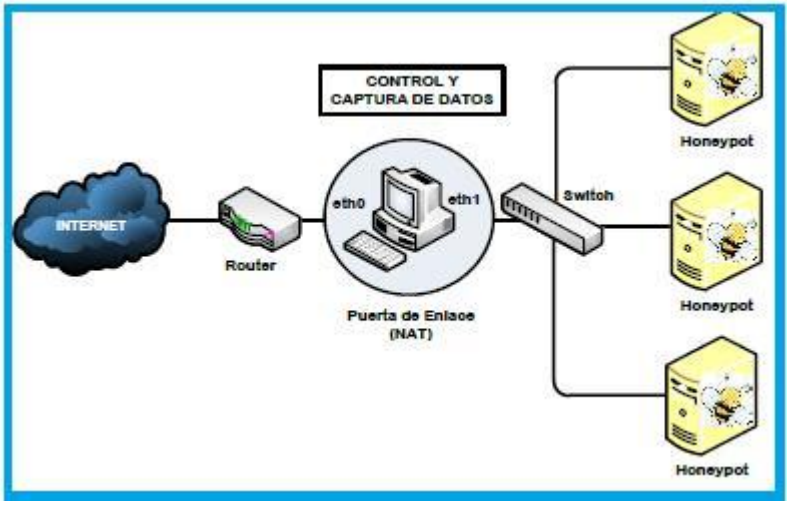

Figura 6. Arquitectura general de una Honeynet de Generación I, Fuente: **T. A. Vinueza Jaramillo**

2.4.3.1.2 Honeynet de Generación II (GEN II)

Aparece en el año 2002 y se caracteriza por incorporar los mecanismos de control y captura de datos en un único dispositivo de capa dos trabajando en modo puente, conocido como Honeywall, que no modifica los paquetes de la red mientras se procesan, ni reduce el tamaño del tiempo de vida, de modo que no se genera ningún tipo de tráfico perceptible por los intrusos.

Brinda un control total en cuanto a las conexiones que entran y salen del Honeypot, ya que a diferencia de la arquitectura de primera generación no se limita la cantidad máxima de conexiones salientes posibles, por tanto, provee un alto nivel de interacción con usuarios

malintencionados. Además, no se recurre a la emulación de servicios, puesto que se ejecuta en sistemas operativos y aplicaciones reales.

Para capturar datos en esta generación se recurren a diversos métodos, pues entre más alternativas se tengan para esta tarea más se garantiza la obtención y recopilación de la información, también se añade un IDS, Sistema detector de intrusos a la puerta del Honeywall adaptándose a las reglas del firewall. Cuando se detecta actividad maliciosa, este bloquea o modifica paquetes del mismo tipo en el futuro, evitando que los Honeypots se conviertan en los atacantes de la red. En la ***Figura 7***, el control y captura de datos los manejan el Gateway Honeywall

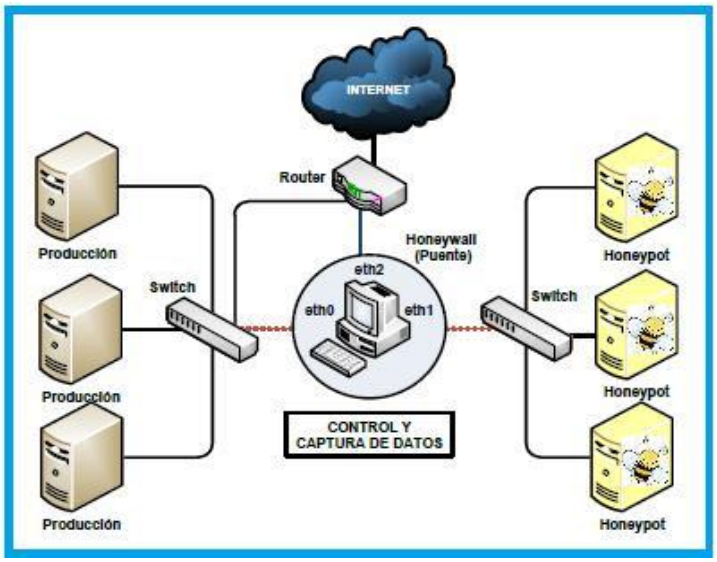

Figura 7. Arquitectura general de una Honeynet de Generación II, Fuente: **T. A. Vinueza Jaramillo**

2.4.3.1.3 Honeynet de Generación III (GEN III)

La tercera generación de las Honeynets apareció en el año 2005. Fundamentalmente, posee la misma arquitectura que la Gen II, pero experimenta ciertas mejoras en cuanto a la capacidad de

gestión y el análisis avanzado de datos. Introduce el concepto de Honeywall Roo, una herramienta Open Source de fácil implantación que integra las funciones de control, captura y análisis de datos.

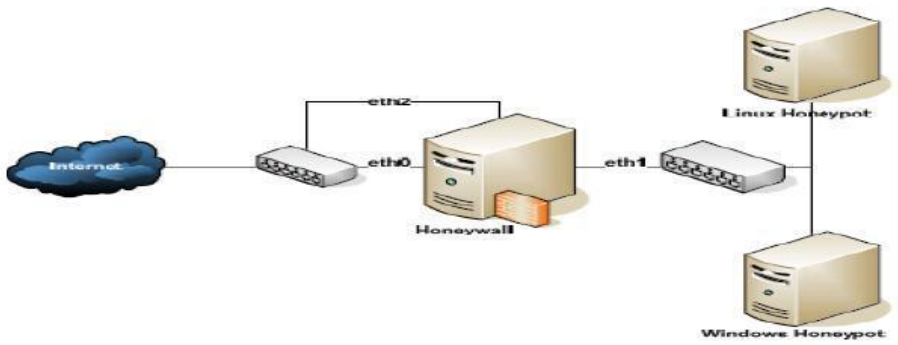

Figura 8. Arquitectura general de una Honeynet de Generación III, Fuente: **S.Y. Dios & D. A. Ortiz**

2.4.3.2 Honeynets Virtuales

Las Honeynets virtuales son una topología que permiten construir una Honeynet completa en una sola máquina física, soportando todos los tipos de generaciones y pudiéndose desarrollar en varios entornos virtuales como VirtualBox o VMware. Son Menor costo, se pueden conectar Máquinas Virtuales para simular una red de computadoras, son fácil de mantener y portables (“Plug and Play”), pero sus contras incluyen: Único punto de fallo y también seguridad limitada en los recursos usados.

Se clasifican en dos tipos: Honeynet Auto Contenida y Honeynet Híbrida.

2.4.3.2.1 Honeynet Auto Contenida

Esta Honeynet se emplea solo en una máquina física para ejecutar toda la Honeynet. Cada sistema operativo contenido dentro de ella actúa independientemente. Su mayor ventaja es el ahorro de costes al minimizar la inversión en recursos físicos, pero debe implementarse en máquinas potentes para que soporten todos los equipos virtuales utilizados.

2.4.3.2.2 Honeynet Híbrida

La Honeynet híbrida incorpora sistemas reales y virtuales. El Honeywall efectúa el control, captura y el análisis de datos en un sistema aislado, mientras que la virtualización de los Honeypots se realiza en un solo equipo. Este tipo de solución aporta seguridad y flexibilidad.

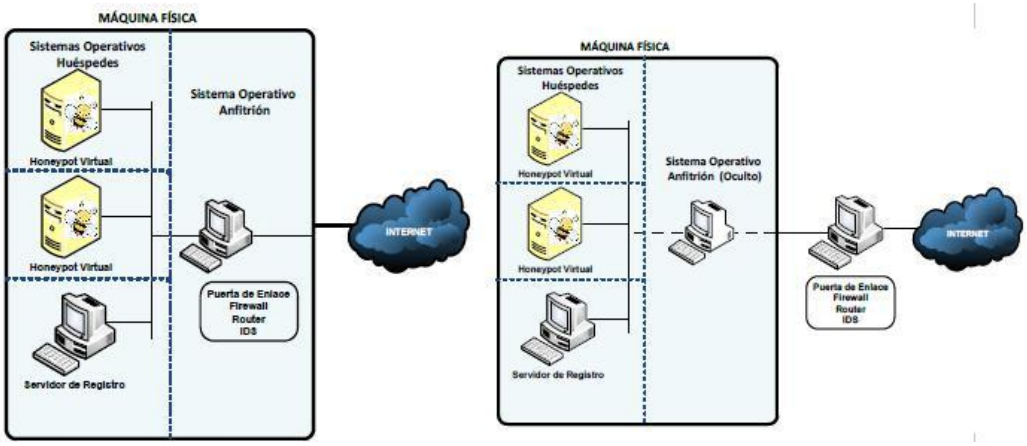

Figura 9. Arquitectura general de una Honeynet Auto contenida (Izq.) y una híbrida (der.),

Fuente: **S.Y. Dios & D. A. Ortiz**

2.5 Marco Legal

Figura 10. Logo de Ministerio de Tecnologías de la Información y Comunicaciones, Fuente: **mintic.gov.co**

Ley 873 de 2004. Por medio de la cual se aprueban el Instrumento de Enmienda a la Constitución de la Unión Internacional de Telecomunicaciones (Ginebra, 1992), con las enmiendas adoptadas por la Conferencia de plenipotenciarios (Kyoto, 1994) (Enmiendas adoptadas por la Conferencia de Plenipotenciarios (Minneápolis, 1998), firmado en Minneápolis, el seis (6) de noviembre de mil novecientos noventa y ocho (1998), y el Instrumento de Enmienda al Convenio de la Unión Internacional de Telecomunicaciones (Ginebra, 1992), con las enmiendas adoptadas por la Conferencia de Plenipotenciarios (Kyoto, 1994) (Enmiendas adoptadas por la Conferencia de Plenipotenciarios (Minneápolis, 1998), firmado en Minneápolis, el seis (6) de noviembre de mil novecientos noventa y ocho (1998).

Ley 1273 de 2009. "De la protección de la información y de los datos". Congreso de la república. Por medio de la cual se modifica el Código Penal, se crea un nuevo bien jurídico tutelado -denominado "de la protección de la información y de los datos"-y se preservan

integralmente los sistemas que utilicen las tecnologías de la información y las comunicaciones, entre otras disposiciones.

Ley 1341 de 2009. Por la cual se definen principios y conceptos sobre la sociedad de la información y la organización de las tecnologías de la información y las comunicaciones.

Artículos 12 y 68 Reglamentados por el Decreto 2044 del 19 de septiembre de 2013. Ley reglamentada por el Decreto 2693 del 21 de diciembre de 2012. Parágrafo 2° del artículo 57 modificado por el artículo 59 de la Ley 1450 de 2011, Inciso 1°y 3° y el parágrafo 1°y 2° del artículo 69 derogado por el artículo 276 de la Ley 1450 de 2011.numerales 6 y 7 del artículo 18, el numeral 11, del artículo 28 y el artículo 29 de la Ley 1341 de 2009 derogados por el Decreto 4169 de 2011.

Capítulo 3. Metodología

3.1 Tipo de Investigación

(Creswell, 1997) y Reichardt (2004) llaman a los experimentos estudios de intervención, porque un investigador genera una situación para tratar de explicar cómo afecta a quienes participan en ella en comparación con quienes no lo hacen.

Teniendo en cuenta el contenido, estructura y finalidad del proyecto investigativo cabe destacar que este será de tipo experimental, puesto que la esencia de la concepción de experimento es que requiere la manipulación intencional de una acción para analizar sus posibles resultados. Siendo este proyecto de índole experimental, donde la manipulación de las variables y casos planteados para el estudio van hacer en entornos controlados por una estructura tecnológica, en donde se efectuarán procedimientos concretos con el fin de observar el cambio que puedan surgir en las variables como velocidad, seguridad, calidad de servicio e integridad de datos, de tal manera que la variable independiente resulta ser de gran interés para el investigador, ya que teóricamente será una de las causas que producen el efecto supuesto a fin de extraer generalizaciones significativas que contribuyan al conocimiento.

Dada la naturaleza de la investigación, esta tendrá un enfoque cuantitativo ya que se pretende analizar parámetros como la seguridad, cantidad de ataques, tiempo y asertividad en la entrega de paquetes, que pueden representarse en valores numéricos y sobre los cuales se puede llevar una evaluación cuantitativa.

A continuación se expone la estructura del marco metodológico en donde se presentan los objetivos de la investigación y con ello cada una de las actividades que contribuirán al desarrollo de este:

Objetivo 1: Caracterizar vulnerabilidades y ataques más comunes en redes cableadas IPv6 que permitan identificar patrones en procedimientos de ataques.

1.1 Revisión documental de los ataques realizados bajo IPv6.

1.2 Análisis de la adaptabilidad de los Honeypots al protocolo IPv6

1.3 Definición y especificación del software a utilizar para la red Honeynet.

1.4 Selección del software correspondiente.

Objetivo 2: Implementar un modelo de red Honeynet estática diseñado para el laboratorio de redes y telecomunicaciones de la Universidad Francisco de Paula Santander Ocaña.

2.1 Diseño topológico de red Honeynet en ambiente IPv6.

2.1 Diseño lógico de la estructura de los componentes Honeynet.

2.3 Implementación de una red Honeynet bajo IPv6.

Objetivo 3: Evaluar el funcionamiento de la red Honeynet, analizando métodos y técnicas de atacantes obtenidos a través de esta, para plantear medidas de aseguramiento.

3.1 Simulación de ataques programados a la red.

3.2 Aplicación del diseño en entorno real de una red Honeynet.

3.3 Análisis evaluativo del funcionamiento de la red Honeynet.

3.2 Población Y Muestra

3.2.1 Población

La población está conformada por el coordinador del laboratorio de redes y telecomunicaciones de la Universidad Francisco de Paula Santander Ocaña, los estudiantes ejecutores del proyecto y los atacantes que intenten penetrar a la Honeynet.

3.2.2 Muestra

Considerando que la población que se verá envuelta en esta investigación será el coordinador del laboratorio de redes y telecomunicaciones de la Universidad Francisco de Paula Santander Ocaña, los estudiantes ejecutores del proyecto y los atacantes el cual se desconoce el número, se tomará como muestra toda la población involucrada en el proceso.

3.3 Técnicas e Instrumentos de Recolección de la Información

En la investigación es necesario aplicar la técnica de observación estructurada ya que permite probar una hipótesis o cuando se quiere hacer una descripción sistemática de algún fenómeno. El instrumento mediante el cual se va a obtener la información para aplicar esta técnica es una ficha de observación para los laboratorios que se van a realizar durante el desarrollo del presente estudio.

3.4 Técnicas de Procesamiento y Análisis de la Información

Se realizará cuando la ficha de observación estructurada contenga los resultados de laboratorio, los cuales se generarán durante la ejecución de la investigación como se muestra en el cronograma de actividades.

.

Capítulo 4. Presentación de resultados

4.1 Caracterizar Vulnerabilidades y Ataques más Comunes en Redes Cableadas IPv6 que Permitan Identificar Patrones en Procedimientos de Ataques.

Tabla 1.

Capacidad de Detección

CAPACIDAD DE DETECCIÓN	
Grupo I de THC-IPv6. Anuncio de Router falso o redirección ICMPv6	
• *fake_router6* Se anuncia como Router e intenta convertirse en el Router predeterminado.	Sintaxis: fake_router6 [-HFD] interface network-address / prefix-length [dns- server [router-ip-link-local [mtu [mac-address]]]] Si se proporciona una dirección de MAC local o de enlace no existente, esto da como resultado un ataque DOS. La opción -H agrega salto por salto (hop by hop), -F encabezado de fragmentación y -D cabecera de destino grande.
• *flood_router6* Inunda la red local con anuncios de Router.	Sintaxis: flood_router26 [-HFD] [-s] [-RPA] interface Cada paquete contiene 17 prefijos y entradas de ruta -F / -D / -H añadido de fragmentos / destino / encabezado salto por salto para eludir la seguridad de RA. -R solo envía entradas de enrutamiento, no información de prefijo. -P solo envía información de prefijo, no entradas de enrutamiento. -A es como -P pero implementa un ataque de George Kargiotakis para deshabilitar las extensiones de privacidad. La opción -s usa vidas pequeñas, lo que resulta en un impacto más devastador.
• *kill_router6* Anuncia que es un destino de un Router para eliminarlo de las tablas de enrutamiento.	Sintaxis: kill_router6 [-HFD] interface router-address [source-mac [destiny-mac]] Si proporciona un '*' como dirección de Router, esta herramienta husmea la red en busca de cualquier paquete de RA e inmediatamente envía el paquete de destrucción. La opción -H agrega salto por salto, -F encabezado de fragmentación y -D cabecera de destino grande.
• *redir6* Implanta una ruta en la IP de la víctima, que redirige todo el tráfico a la IP blanco.	Sintaxis: interface redir6 víctim-ip target-ip original-router new-router [new -router-mac] [hop-limit] Debe conocer el Router que manejaría la ruta. Si la nueva MAC del Router (new router mac) no existe, esto da como resultado un ataque DOS. Si el TTL del objetivo no es 64, entonces especifique que esta es la última opción.

Grupo II de THC-IPv6. Falso Anuncio / Solicitud de Vecino	
• *dos-new-ip6* Esta herramienta evita que surjan nuevas interfaces IPv6.	Sintaxis: interfaz dos-new-ip6 Esta herramienta evita que surjan nuevas interfaces IPv6, enviando respuestas para duplicar cheques ip6 (DAD). Esto da como resultado un DOS para nuevos dispositivos IPv6.
• *parsite6* Este es un "Spoofer (falsificación y suplantación de identidad) de ARP" para IPv6.	Sintaxis: parasite6 [-lRFHD] interface [fake-mac] Esto es un Spoofer de "ARP" para IPv6, redireccionando todo el tráfico local a su propio sistema (si fake-mac (MAC falsa) no existe) respondiendo falsamente a las solicitudes de vecinos. Opción -l hace bucles y reenvía los paquetes por destino cada 5 segundos. La opción -R también intentará inyectar el destino de la solicitud de derivación de seguridad NS: -F por fragmentos, -H salto-por-salto y -D cabecera de destino grande.
• *fake-advertise6* Anuncia la dirección IPv6 en la red.	Sintaxis: fake_advertise6 [-DHF] [-Ors] [-n count] [- w segundos] interface announced-ip-address [destiny-address [announced-mac-address [ip-address]]] Anuncia la dirección IPv6 en la red (con su propia MAC si no se especifica), enviándola a todos los nodos dirección de multidifusión si no se establece una dirección de destino. La dirección IP de origen es la dirección anunciada si no está configurada. Opciones de envío: -n count envía cuántos paquetes. (valor predeterminado: infinito) -w segundos de espera entre los paquetes enviados. (valor predeterminado: 5) Opciones de indicador: -O NO establece el indicador de reemplazo (predeterminado: activado) -r indicador de Router (predeterminado: desactivado) -s DO establece el indicador solicitante (valor predeterminado: desactivado) - ND Opciones de evasión de seguridad (se pueden combinar): -H agrega un encabezado salto por salto -F añade un encabezado de fragmento de un disparo (se puede especificar varias veces) -D añade un encabezado de destino que fragmenta el paquete.
• *flood_advertise6* Inunda la red local con anuncios de vecinos.	Sintaxis: flood_advertise6 interface Inunda la red local con anuncios vecinos.
• *flood_solicitate6* Inunda la red con solicitudes de vecinos	Sintaxis: flood_solicitate6 interface [target] Inunda la red con solicitudes de vecinos.

Grupo III de THC-IPv6. Falso DHCPv6 / Servidor DNS o Cliente	
• *flood_dhcpc6* Flooder de cliente DHCP.	Sintaxis: flood_dhcpc6 [-n | -N] [-1] [-d] interface [domain-name] DHCP cliente flooder. Úselo para reducir el grupo de direcciones IP que ofrece un servidor DHCPv6. . Nota: si el grupo es muy grande, esto no tiene sentido. Por defecto, la dirección MAC de IP local del enlace es aleatoria, sin embargo, esto no funcionará en algunas circunstancias. -n utilizará el MAC real, -N el MAC real y la dirección local de enlace. -1 solo mostrara una dirección pero no la solicitará. Si no se usa -N, debe ejecutar parásito6 en paralelo. Use -d para forzar las actualizaciones de DNS, puede especificar un nombre de dominio en la línea de comandos.
Grupo IV de THC-IPv6. Consumidor maligno de uso de Red o uso de CPU	
• *rsmurf6* Smurfea la red local de la víctima.	Sintaxis: interfaz rsmurf6 víctim-ip Smurfea la red local de la víctima. Nota: esto depende de un error de implementación, actualmente solo verificado en Linux. Evil: "ff02::1" como víctima pondrá DOS en su LAN local completamente, es una versión multi-hilo de smurf6.
• *smurf6* Smurfea la red local de la víctima.	Sintaxis: interfaz rsmurf6 víctima-ip Smurfea de la red local de la víctima. Nota: esto depende de un error de implementación, actualmente solo verificado en Linux. Evil: "ff02::1" como víctima pondrá DOS en su LAN local completamente
• *sendpees6* Enviar mensajes de solicitud de vecinos (SEND).	Uso: sendpees6 <inf> <key_lenght_prefix> <prefix> <victim> Envía mensajes de solicitud de vecino SEND y hace un objetivo para verificar una lota firmas CGA y RSA
• *sendpeesmp6* Enviar mensajes de solicitud de vecinos (SEND).	Uso: sendpeesmp6 <inferface> <key_length> <prefix> <victim> Envía mensajes de solicitud de vecino (SEND) y hace un objetivo para verificar una lota firmas CGA y RSA Ejemplo: sendpeesmp6 eth0 2048 fe80:: fe80::1

Fuente: **KALITOOLS**

Capacidad de detección

6Guard puede detectar la mayoría de los ataques iniciados por THC-IPv6 y los métodos avanzados de descubrimiento de host IPv6 utilizados por Nmap . En términos prácticos, esta versión beta podría detectar los ataques de la siguiente manera:

Grupo I de THC-IPv6. Anuncio de Router falso o redirección ICMPv6

fake_router6: es un Exploit que hace parte del kit de herramienta de van Hauser's IPv6 que permite al atacante anunciarse como un Router en la red con la más alta prioridad. Inclusive si otros Router IPv6 están presentes en la red, los nuevos clientes se conectan al Router de enrutamiento que crea fake_router6. Esta herramienta (fake_router6) que envía mensajes RA (mensaje de anuncio de Router) fragmentados; se compone de los siguientes parámetros necesarios para su ejecución: (Sánchez, 2015)

- La interfaz donde se enviaran los mensajes RA
- La dirección de enlace local donde que se utilizara como dirección IPv6 de origen
- El prefijo 2001: db8: bad: bad ::/ 64
- El MTU, 1000 en este caso (por lo general se establece en 1500 en los mensajes RA normales).

Del fake_route6 muestra la salida que indica la métrica de 16 para la ruta predeterminada mediante la computadora deshonesta, una métrica de 256 para la ruta predeterminada mediante el Router predeterminado legitimo cuya dirección de enlace local es FE80::1. El ataque fake_route6 también puede ajustar la dirección de enlace local, MTU y la dirección. (Barker, 2013)

Flood_Router6 *Inunda* la red local con anuncios de Router. Un atacante remoto puede causar una denegación de servicio (consumo de CPU y bloqueo del sistema) al enviar varios mensajes de anuncio de Router (RA) con diferentes direcciones de origen. Con el paso inevitable a IPv6, este problema que se conoce desde hace casi un año y se está volviendo cada vez más crítico. El problema parece ser que Microsoft y otros proveedores de IPv6 no ofrecen mucho en cuanto a soluciones. (CYBORG, 2015)

Según (Hacker, 2017) El problema es que la actualización de las tablas de enrutamiento y la configuración de direcciones IPv6 requiere muchos recursos de CPU (es decir, 100%). Si una red está inundada de anuncios de Router aleatorios, Windows (y otros sistemas operativos como FreeBSD) tienen dificultades para actualizar sus tablas de enrutamiento. La denegación de servicio permanece en vigencia hasta que finaliza la inundación.

kill_router6: Este ataque anuncia que el Router destino está mal para eliminarlo de las tablas de enrutamiento. Conjuntamente si se adiciona un '*' como dirección de Router, esta herramienta inspecciona toda la red para detectar cualquier mensaje RA (Mensaje de anuncio de Router) para así generar y enviar el paquete de destrucción.

Problemas de redirección (redir6)

La función de redireccionamiento de mensajes IPv6 es un mecanismo permite a un dispositivo enviar mensajes de redireccionamiento IPv6 vecino del Protocolo de mensajes de control de Internet (ICMP) para informar a los hosts de los mejores nodos de primer salto (dispositivos o hosts) en la ruta a un destino. (IPv6, 2017)

Este proceso de redirección se representa en la ***Figura 11*** de a continuación: donde el host A contiene una ruta predeterminada del Router R2 y no tiene conocimiento de que Router R1 tenga

una ruta mejor y más específica para 2001: DB8:2::/64. Por lo tanto, cuando el host A desea enviar un paquete a 2001: DB8:2::1, lo envía a la dirección MAC del Router predeterminado; esto es R2. Cuando R2 recibe este paquete y, al verificar su propia base de información de reenvío, detecta que R1 tiene una ruta mejor, R2 envía inmediatamente un redireccionamiento ICMPv6 al host A con la información de que R1 tiene una mejor ruta a la red 2001: DB8:2::/64. El host A instala esta ruta más específica en su tabla de enrutamiento y comienza a enviar todos los paquetes con una configuración de destino de 2001: DB8:2::/64 al Router R1, logrando una ruta más corta al destino. (Hogg, IPv6 Security, 2008)

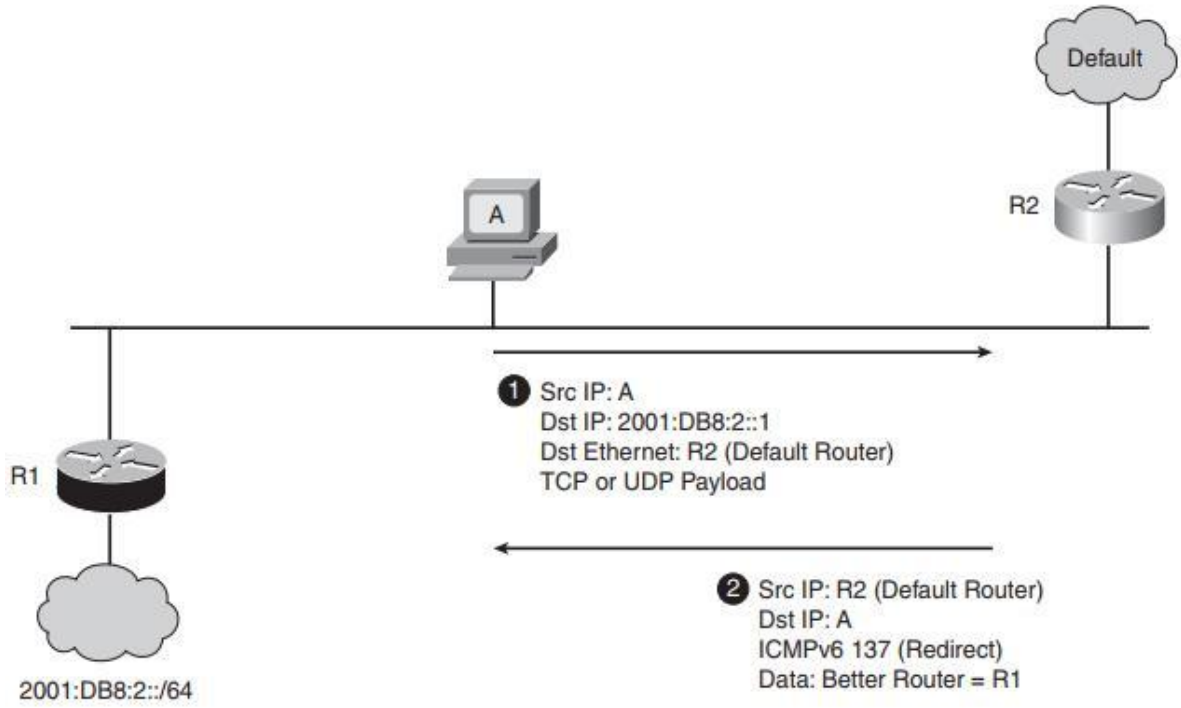

Figura 11. Redireccionamiento del tráfico con Redirect ICMPv6, Fuente: **Scott Hogg**

Sin embargo en este procesamiento no existe una fase de verificación inherente a la redirección ICMPv6, dado que este mecanismo es bastante simple, se debe incluir una copia del paquete que causa la redirección en el mensaje de redirección ICMPv6. A pesar de ello el

atacante no puede enviar a ciegas un mensaje de redirección ICMPv6 ya que este debe tener acceso al contenido del primer paquete.

Sin embargo en el kit de herramientas de THC-IPV6 tiene una herramienta llamada redir6 que implanta una ruta src-ip (dirección de origen) que redirige todo el tráfico a la tarjeta IP. Este debe conocer el Router que manejara a ruta, si el new-router-mac (nueva MAC del Router) no existe, esta emite como resultado un DOS.

A continuación se muestra el procedimiento para ejecutar el ataque redir6, con respecto al ejemplo anterior, ver la ***Figura 11*** anterior:

1. Se envía una solicitud de eco ICMPv6 a la víctima, que es A en la ***Figura 11*** anterior, con una dirección de origen falsificada (por ejemplo, 2001: DB8:2::).
2. El atacante puede adivinar que la víctima A, enviará una respuesta de eco ICMPv6 a 2001: DB8:2::1, ya que sabe exactamente lo que A enviará.
3. El atacante puede enviar el redireccionamiento ICMPv6 con la dirección de origen forjada del Router predeterminado y que contiene una copia de la respuesta de eco supuesta ICMPv6

Según IPv6 Security este ejemplo se vuelcan los contenidos de un caché de ruta de host de Windows XP para la dirección 2001:4860:0:1001::68 (que es IPv6.google.com) en la interfaz 7 con el comando IPv6 rc; un Traceroute también muestra los primeros tres Router en el camino a este host. (Hogg, IPv6 Security, 2008)

Grupo II de THC-IPv6. Falso Anuncio / Solicitud de Vecino

dos-new-ip6: Esta herramienta evita que surjan nuevas interfaces IPv6, enviando respuestas a duplicar cheques IP6 (DAD). Esto da como resultado un DOS para nuevos dispositivos IPv6. En la siguiente ***Figura 12*** se evidencia el funcionamiento del ataque dos-new-ip6:

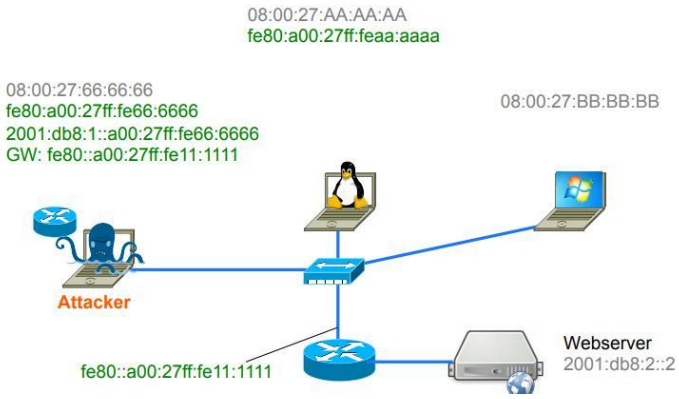

Figura 12. Detección de dirección duplicada, Fuente: **F. Herberg IPv6 Scurity**

Dos-new-ip6 percibe el nuevo IPv6 en beneficio de los procesos de SLAAC (Configuración automática de dirección sin estado) NDP, con el propósito de rehusar o eliminar nuevas direcciones IPv6. DoS-new-IPv6 también utiliza Detección de direcciones duplicadas (DAD) para realizar un DoS en cualquier host que busque crear una nueva dirección IPv6. Su simplicidad del suceder con solo emitir un mensaje, señalando que la dirección seleccionada por ese host ya está tomada o en uso, de vuelta al host que intenta crear una nueva dirección. -------- Parafraseado de (Hogg, IPv6 Security, 2008)

Parasite6: es un tipo de Spoofer que en términos de seguridad informática se refiere a "hacerse pasar por otro" donde el atacante se vale de técnicas o suplantación de identidad.

Técnicas que permiten el envenenamiento de ARP redirigiendo todo el tráfico de datos locales al sistema del atacante respondiendo falsamente a solicitudes el cual le delega a estas funciones inapropiadas ante el sistema.

El atacante puede utilizar la herramienta **parasite6** para enviar una respuesta afirmando que la MAC solicitada se corresponde con su dirección IP, en el siguiente esquema se muestra el comportamiento del ataque.

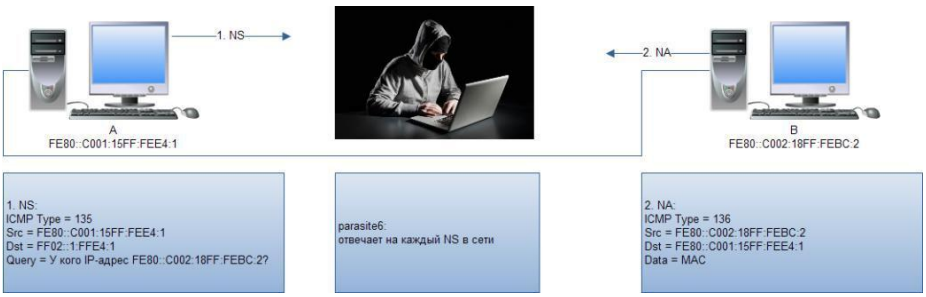

Figura 13. Ataque Parasite6, Fuente: **hackmag.com**

De hecho cualquier ataque que sea capaz de insertarse en una red entre dos host que a su vez le permita acceder y manipular información que transita por la red sin consentimiento de los host, a estos se les incluye dentro de los mencionados Man in the Middle (MITM, Hombre en el Medio).

En similitudes parasite está presente en IPV4 e IPV6, pues no se han alterado fundamentalmente. De hecho, hay muchas nuevas oportunidades para los ataques MITM en IPv6, por ejemplo, un atacante puede crear un Router falso utilizando anuncios de Router (RA). Para este ataque usamos Parásito6. (Hogg, IPv6 Security, 2008)

fake-advertise6 El acceso no autorizado se refiere a la clase de ataques donde el adversario intenta explotar la política de transporte abierto inherente al protocolo IPv4. Nada en la pila del

protocolo IP limita el conjunto de hosts que pueden establecer conectividad con otro host en una red IP. Los atacantes se basan en este hecho para establecer la conectividad con protocolos y aplicaciones de capa superior en dispositivos de interconexión de redes y hosts finales. (Convery & Miller, 2004)

De tal manera que cuando un atacante obtenga un acceso no autorizado a la red, este podría causar daños de muchas maneras, desde acceder a archivos confidenciales, incrustando virus u obstaculizando el rendimiento de la red al inundar su red con paquetes ilegítimos. En los mencionados ataques de acceso no autorizado Fake Advertise6 es ubicado en este grupo, ya que permite al usuario anunciar falsamente una dirección IPV6 en la red (con su propia MAC, si no está definido) con ayuda del protocolo Neighbor Advertisement (NA, Anuncio de Vecino) enviándola a la dirección de multidifusión de todos los nodos si no se especifica ningún objetivo. De este modo determinar las direcciones de las otras capas de enlace, encontrar Router y mantener información de accesibilidad sobre las rutas a los vecinos activos.

El NA falso puede enviarse a un objetivo específico o a la dirección de multidifusión de todos los nodos. (Gehrke, 2015)

flood_advertise6 Este ataque se comparta inundando dispositivos de la red (Router o Host) con gran cantidad de anuncios para traficar sobre esta, trayendo como consecuencia el no poder procesar dicho tráfico en la red lo que llevara a no estar disponible o fuera de servicio. Este ataque es muy frecuente tano en IPv4 como en IPv6 debido a que los principios básicos del ataque de inundación siguen siendo los mismos. Un ataque de inundación puede ser local o ataque distribuido de denegación de servicio (DDoS), cuando el dispositivo de red objetivo está siendo inundado por el tráfico de red de muchos hosts simultáneamente.

Parafraseado de: (Durdağı & Buldu, 2010)

Los nuevos tipos de encabezados de extensión en IPv6, los nuevos tipos de mensajes ICMPv6 y la dependencia de las direcciones de multidifusión en IPv6 (por ejemplo, todos los Router deben tener direcciones de multidifusión específicas del sitio) pueden proporcionar nuevas formas de uso indebido en los ataques de inundación.

Citado este párrafo de: (Gehrke, 2015)

flood_solicitate6 *Inunda* la red con solicitudes vecinas.

Grupo III de THC-IPv6. Falso Servicio de DHCPv6 / DNS o Cliente

flood_dhcpc6

DHCPv6 depende de la comunicación UDP sin estado utilizando los puertos UDP 546 y UDP 547. Esto hace que DHCPv6 sea particularmente vulnerable al ataque falso, en el que los mensajes SOLICIT se generan con prefijos de origen aleatorio. ***Flood_DHCPc6 es un flooder de cliente DHCP.*** Es usado para reducir el grupo de direcciones IP que ofrece un servidor DHCP6 teniendo en cuenta que si el grupo no es muy grande este no tendrá sentido y una subred IPv6 tiene más de 18 quintillones de direcciones

Grupo IV deTHC-IPv6. Consumidor Maligno de Uso de Red o de Uso de CPU

Inundación de paquetes

Las redes IPV4 son susceptibles a los ataques Smurf, de hecho son bastantes simples y a su vez devastadores. Esta técnica consiste en enviar la solicitud de ping mientras se falsifica un paquete desde la dirección de la víctima (la dirección a la que, en teoría, el servidor debe responder) y

proporciona la dirección IP de un equipo de destino. De este modo todos los host pertenecientes la LAN reciben ese paquete con gran carga, a continuación, se emite una petición ICMP (simulando un Ping) a cada una de ellas en serie, varias veces, enviando una respuesta de eco a la dirección de la víctima falsificada.

- El servidor transmite la solicitud a toda la red;
- Todos los equipos de la red envían una respuesta al servidor de difusión;
- El servidor redirecciona las respuestas al equipo de destino.

De este modo, cuando el equipo del atacante envía una solicitud a varios servidores de difusión ubicados en diferentes redes, todas las respuestas de esos equipos se enrutarán al equipo de destino. Véase en la ***Figura 14*** a continuación:

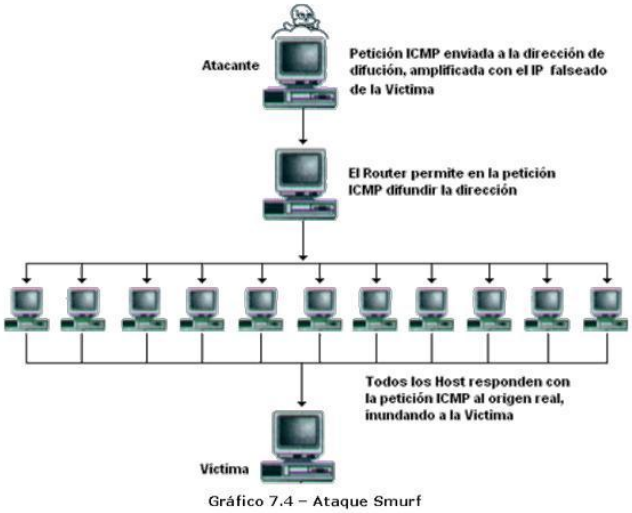

Figura 14. Ataque Smurf, Fuente: **segu-info.com**

Debido a la falta de direcciones de difusión en IPv6, se ha considerado obsoleto y hace que estos ataques sean limitados. En gran parte IPv6 utiliza la multidifusión puesto que los atacantes pueden

aprovechar las direcciones de multidifusión y las pilas de IPv6 mal implementadas. Un atacante en una subred podría intentar enviar tráfico a la dirección de multidifusión de todos los nodos de enlace local (FF02::1) y la dirección de multidifusión de todos los Router de todos los locales (FF02 :: 2). (IPv6, 2017)

El kit de herramientas de THC-IPV6 proporciono algunos ejemplos de amplificación de multidifusión, estas utilidades son dos SMURF6 y RSMURF6. Ambos operan de manera muy similar a los ataques pitufos de IPV4 con la diferencia de que estos usan multidifusión para amplificar el ataque.

Este tipo de ataque sigue siendo muy relativo en la subred local, donde puede generar cargas de tráfico de red y también puede ser viable en subredes remotas que no han implementado IPv6 correctamente (esta es una rara excepción a la regla). El kit de herramientas de THC proporcionó SMURF6 y RSMURF6, FLOOD_ADVERTISE6, FLOOD_DHCPC6 que se utilizaron para representar este tipo de ataque.

smurf6

La herramienta smurf6 envía paquetes de solicitud de eco de Protocolo de mensajes de control de Internet versión 6 (ICMPv6) a la dirección de multidifusión FF02::1, y luego los hosts en esa LAN que son vulnerables al ataque generan paquetes de respuesta de eco ICMPv6 de vuelta a la fuente. Que es la víctima desconocida La víctima smurf6 puede estar en la subred local con el atacante o en una subred remota. (IPv6, 2017)

rsmurf6

El ataque rsmurf6 en relación con smurf6 está codificada de forma un poco diferente. Envía paquetes de respuesta de eco ICMPv6 que se obtienen de FF02::1 y están destinados a equipos remotos. Si la computadora de destino (víctima) es una distribución de Linux que puede responder a paquetes provenientes de una dirección de multidifusión, responde a la fuente, lo que causa una fuga de tráfico en la LAN remota. Esta forma de amplificación es particularmente peligrosa porque cada paquete generado por rsmurf6 se traduciría en numerosos paquetes en la LAN remota. rsmurf6 es como un smurf6 inverso y solo funciona en implementaciones codificadas incorrectamente de la pila IPv6. Por lo tanto, no es tan efectivo como lo era cuando existían sistemas operativos más vulnerables. (IPv6, 2017)

Ambos ataques (SMURF6 y RSMURF6) para la Denegación de Servicio Distribuidos (*DDoS, Distributed Denial of Service*), que actualmente saturan una red de computadoras o un sitio web hasta volverlos inoperantes, seguirán representando una amenaza para las empresas en la nueva versión. Aunque con IPv6 se pueden mitigar los efectos de los ataques DDoS hasta un cierto punto, no los previene, dejando recursos en riesgo de ser bombardeados al punto de ser parados por completo.

Por otro lado tenemos:

sendpees6 Los dispositivos que no son de confianza en una red IPv6 pueden enviar paquetes no autorizados que simulan Router y otros dispositivos en la red, lo que permite varios ataques de hombre en el medio y de denegación de servicio, como se ha mencionado anteriormente. La solución planificada para este problema es Secure Neighbor Discovery (SeND) que asigna a cada

dispositivo una Dirección criptográficamente generada, en lugar de utilizar la dirección MAC o un número aleatorio. La porción de host de la dirección IPv6 (los 64 bits más a la derecha) depende de una firma RSA, y cada dispositivo tiene su propia clave privada.

Por lo tanto, los dispositivos de recepción pueden verificar que los paquetes provienen de la fuente adecuada, realizando cálculos criptográficos.

Este tipo de ataque Envía mensajes de solicitud de vecino SEND (Secure Neighbor Discovery) y establece el objetivo para verificar una lota CGA (dirección generada criptográficamente) y firmas RSA ((Rivest, Shamir y Adleman). supone una carga para la máquina que recibe los. Este ataque DoS desperdicia tiempo de CPU en el objetivo al enviar una gran cantidad de paquetes firmados que debe verificar. El atacante ejecuta el comando. (Si se encuentra en una red aislada, puede atacar todo el segmento de red con una dirección de multidifusión de ff02::1).

./sendpees eth0 1024 dead:: fe80::887:4229:5f43:81c6

Inmediatamente se deberá evidenciar el aumento de la CPU en la máquina de destino. Puede no ir al 100%, pero debería verlo subir.

sendpeesmp6 este ataque es una versión multiproceso de sendpees6.

4.2 Implementar un modelo de red Honeynet estática diseñado para el laboratorio de redes y telecomunicaciones de la Universidad Francisco de Paula Santander Ocaña.

Ubuntu 16.04 LTS, Actualización

Antes de iniciar a realizar cualquier instalación y configuración es importante tener actualizado nuestros sistemas operativos anfitriones y virtualizados, en este caso en el servidor Hewlett Packard, es necesario actualizar el Ubuntu 16.04 LTS para poder realizar futuras instalaciones y configuraciones. En el caso de nuestra Universidad fue necesario liberar el límite de descarga ya que por cuestiones de seguridad está limitado, acción realizada por La División De Sistemas.

Mediante "Software de Ubuntu"

Podemos Hacerlo primero mediante la aplicación "**Software de Ubuntu"** e instalando todas las actualizaciones disponibles que este nos ofrece, pero con prioridad **"Actualizaciones del SO"** y **"Actualizaciones de software"**:

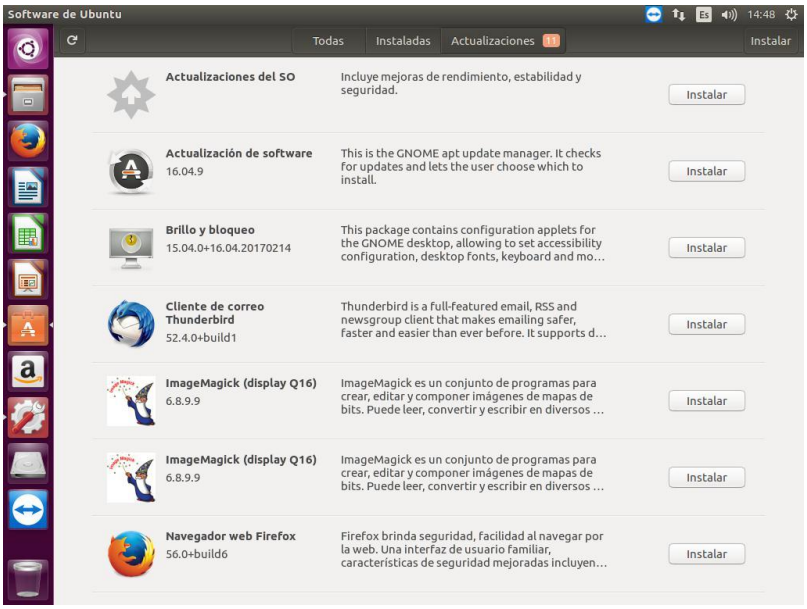

Figura 15. "Actualizaciones del SO" y "Actualizaciones de software", Fuente: **Autores del Proyecto**

Luego de completarlas este cuadro también nos saldrá en forma de pop out:

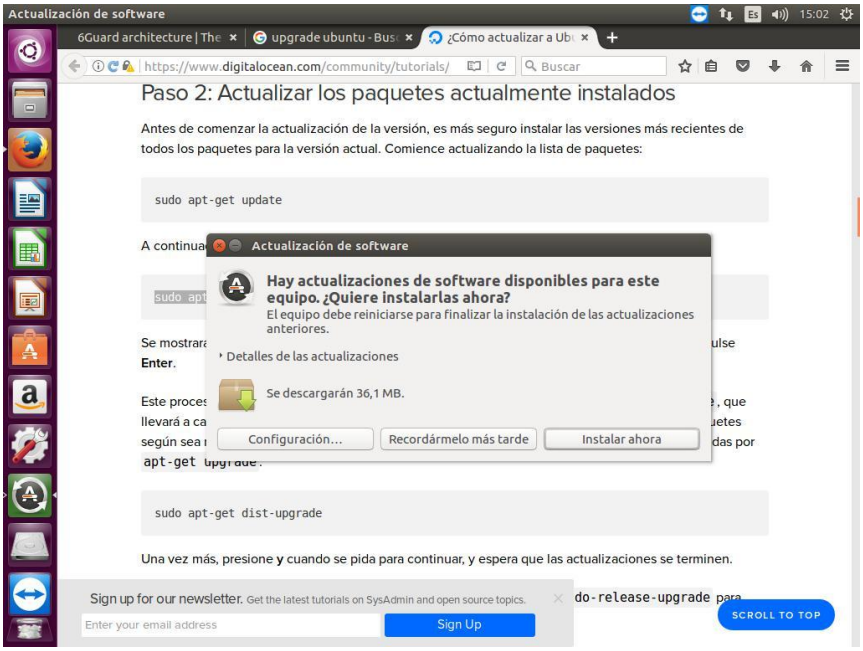

Figura 16. Actualización de Software, Fuente: **Autores del Proyecto**

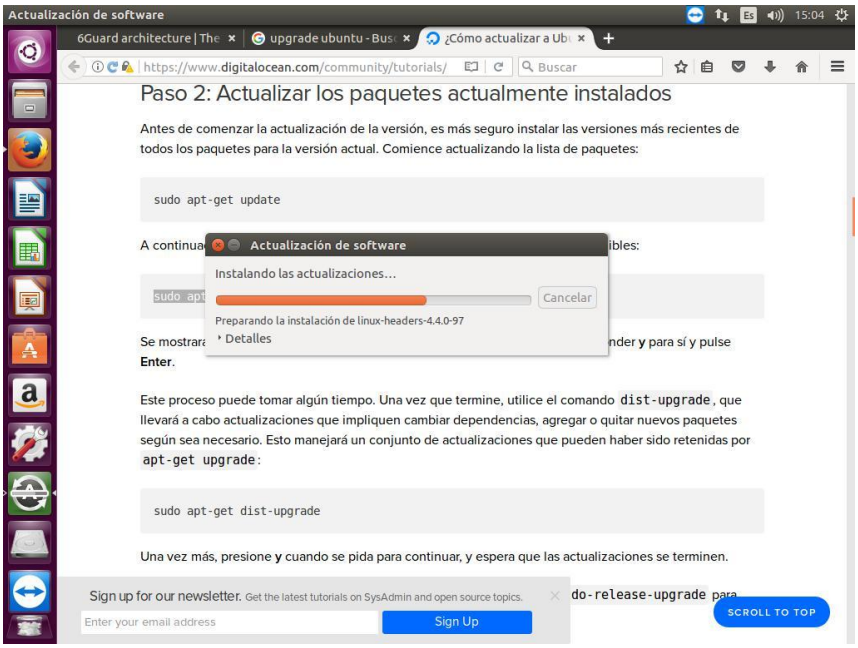

Figura 17. Ejecución de Actualización de Software, Fuente: **Autores del Proyecto**

Luego de reiniciar, y revisar mente nuevamente la aplicación debería salir la siguiente pantalla:

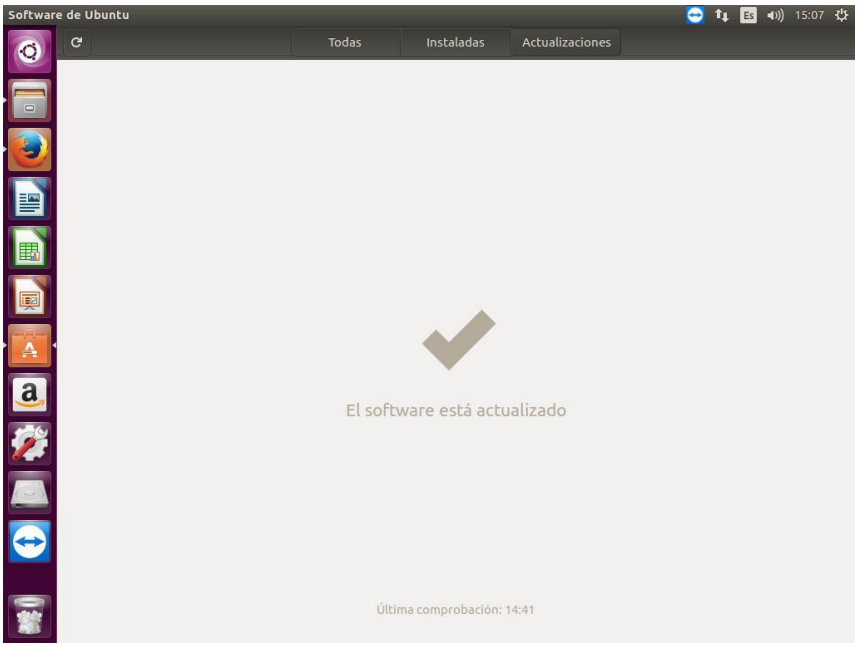

Figura 18. Software Actualizado, Fuente: **Autores del Proyecto**

Mediante La Terminal

Para estar totalmente seguros es necesario actualizar también el sistema operativo por medio de la terminal. Siempre se utiliza el comando ***sudo su***, Para entrar por medio del usuario **Root** para que así se tenga acceso a todas las rutas y dependencias del sistema operativo que necesitan modificarse. Utilizaremos el comando ***sudo apt-get update*** para empezar a actualizar todos los archivos de configuración, y le daremos ***S*** cuando se nos pregunte si deseamos continuar:

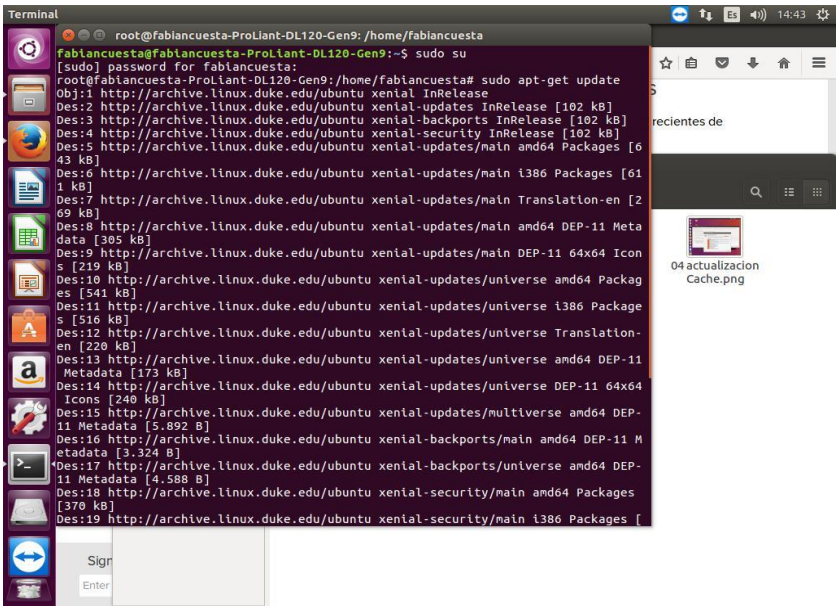

Figura 19. Actualización mediante la terminal, Fuente: **Autores del Proyecto**

Una vez finalizado se procederá a utilizar el comando ***sudo apt-get upgrade*** para comenzar a instalar las propiamente dichas actualizaciones:

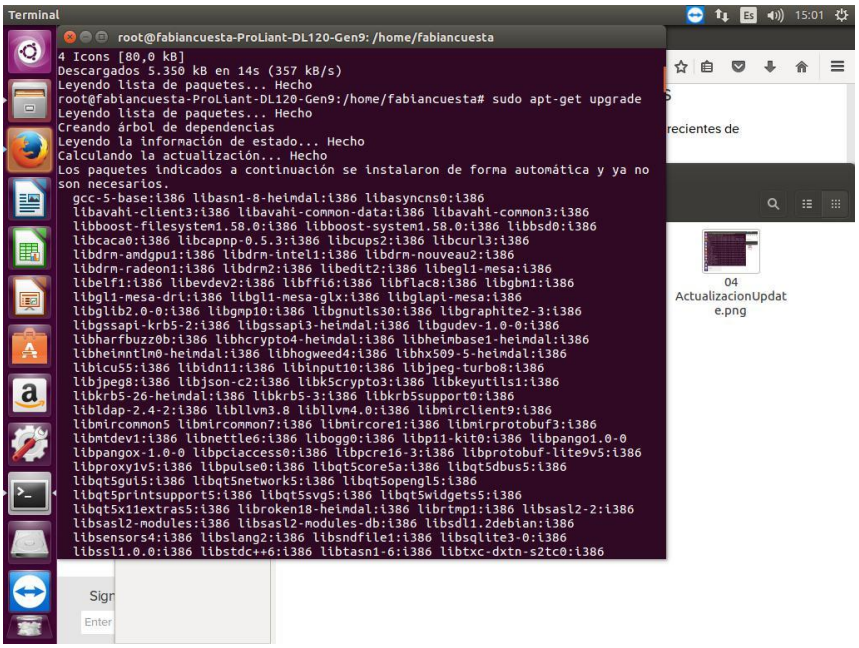

Figura 20. El comando ***sudo apt-get upgrade,*** Fuente: **Autores del Proyecto**

Confirmaremos otro tipo de actualizaciones importantes con el comando: ***sudo apt-get dist upgrade***

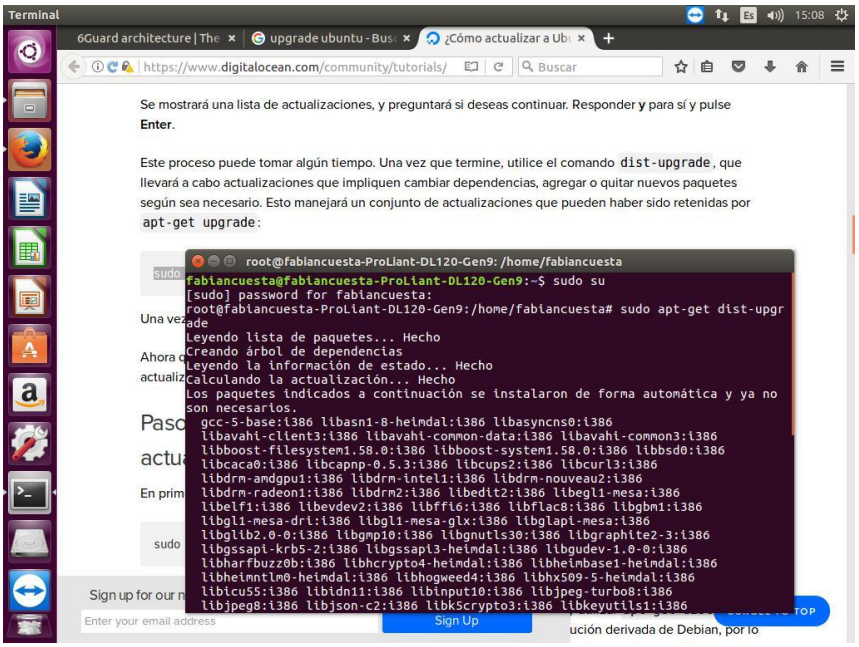

Figura 21. El comando: ***sudo apt-get dist-upgrade***, Fuente: **Autores del Proyecto**

Cuando acabe, nuestro sistema operativo Ubuntu quedara listo para empezar a instalar los componentes de la Honeynet.

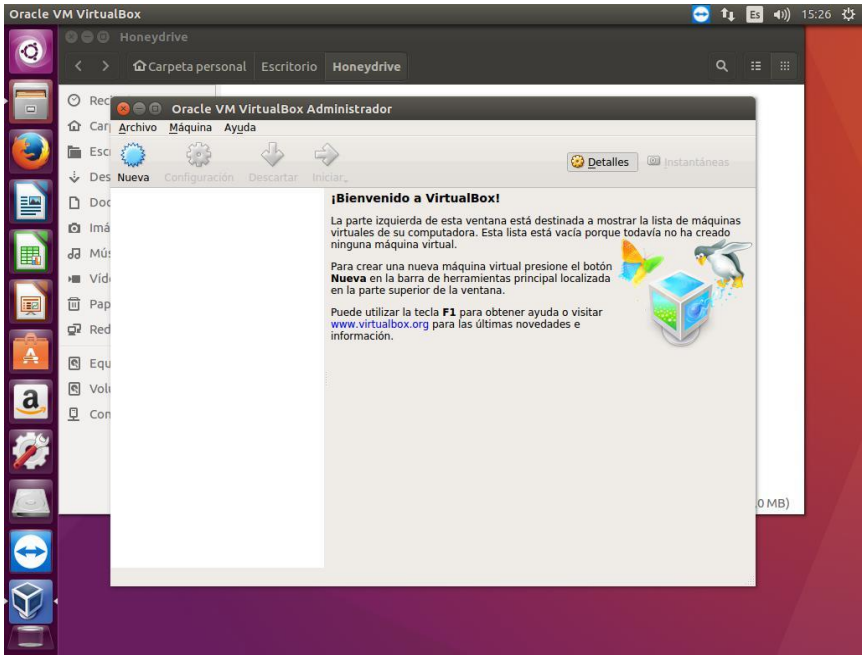

Figura 22. Instalación de Oracle Virtualbox, Fuente: **Autores del Proyecto**

Instalación de Oracle VirtualBox

Virtualbox es un software clave para crear máquinas virtuales e la Honeynet, las cuales siempre van a ser utilizadas para implementar los diferentes elementos que componen una Honeynet, Al igual que lo anterior, Virtualbox se puede instalar de las dos formas, aunque es más aconsejable siempre hacerlo por la terminal.

Mediante “Software de Ubuntu”

También se puede conseguir bajándolo desde su página para la versión del sistema operativo que se está utilizando:

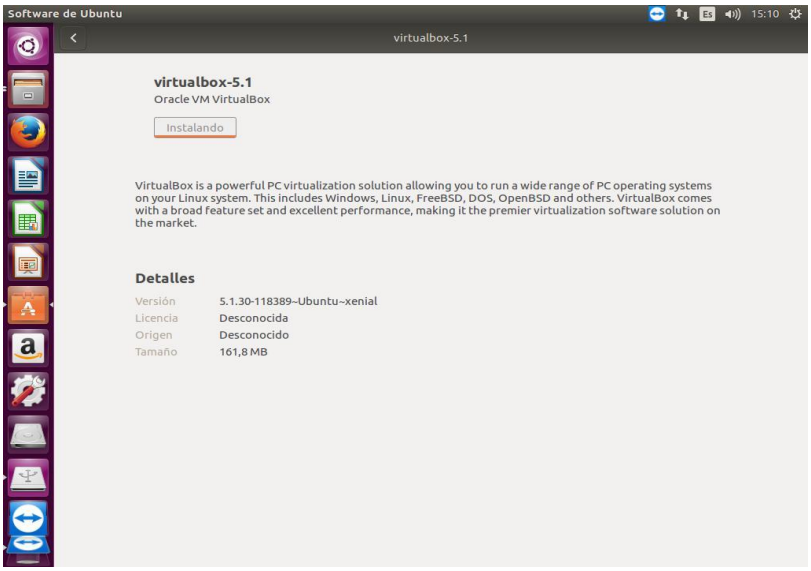

Figura 23. Actualización de Virtualbox mediante "Software de Ubuntu", Fuente: **Autores del Proyecto**

Mediante La Terminal

Con el uso del comando para su versión 5.0 la cual es la más cómoda y recomendable **sudo *apt-get install virtualbox-5.0,*** para este caso se reemplazó la versión 5.1 instalada por el método anterior:

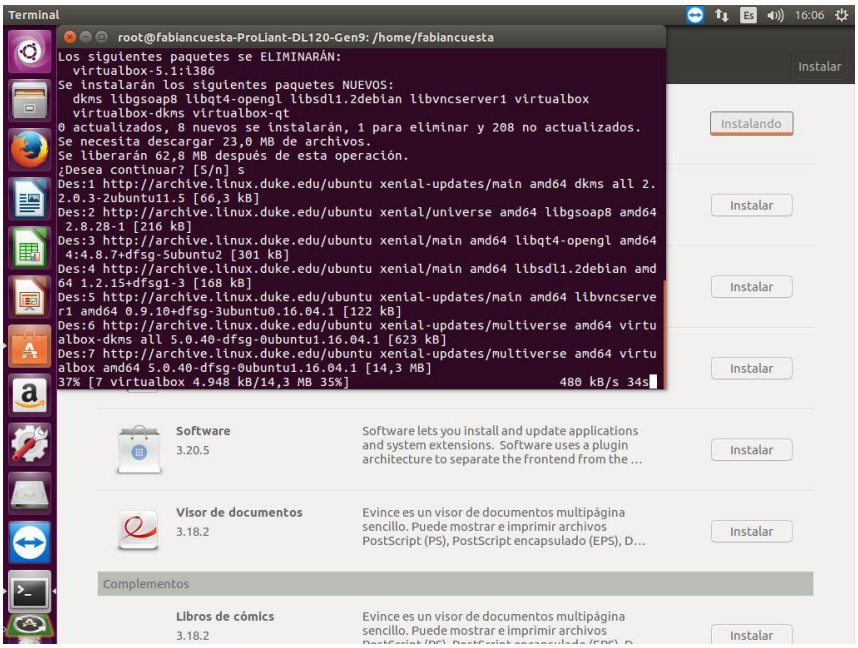

Figura 24. Actualización de Virtualbox Mediante La Terminal, Fuente: **Autores del Proyecto**

Honeynet IPv6 de prueba

La topología IPv6 de prueba que se implementó a continuación sirvió para verificar la funcionalidad pasiva (sin un host atacante aun, conectado al Router y a la escucha) del Honeypot 6Guard, el objetivo de esta prueba es hacer la conexión del 6Guard con el Router Cisco. Su funcionalidad completa se probara después al implementarse la Honeynet final con el Atacante incluido y los ataques registrados.

Aplicación de Honeypots IPv6 en entorno LAN

Como ya se ha especificado anteriormente hay más que todo 2 tipos de Honeypots: los de alta y los de baja interacción. Debido al alto conocimiento y uso que se tiene de IPv4 resulta que todo lo que se desarrolla e implementa para este protocolo es de alta interacción y generalmente

se utiliza en entornos de internet. Como IPv6 es tan complejo y abundante, el uso de Honeypots para internet resulta ineficaz, solamente piénsese bajo el punto de vista de un ataque de inundación de IPs, suponiendo que una computadora con suficiente potencia y el software necesario pudiera escanear 1 millón de direcciones por segundo, la exploración de toda la subred llevaría casi 500.000 años [2]. Es por esto que todo lo que relacionado en IPV6 en cuanto a Honeynets se realiza en entornos de intranet, y los Honeypots IPv6 (Que son muy pocos en relación a los de IPv4) suelen ser de baja interacción. Es por esta razón que se eligió realizar la Honeynet IPv6 en un entorno LAN.

El protocolo IPv6 está diseñado para que sea resistente a los ataques de enlace local, pero los mecanismos de prevención correspondientes (por ejemplo, IPSec, SEND, RA-Guard) se han implementado rara vez en sistemas operativos de punto final en la actualidad. Incluso su implementación en Router y otros elementos de red activos está lejos de ser óptima a pesar del hecho de que algunos productores han implementado ciertos mecanismos. (Sochor & Zuzcak, 2015)

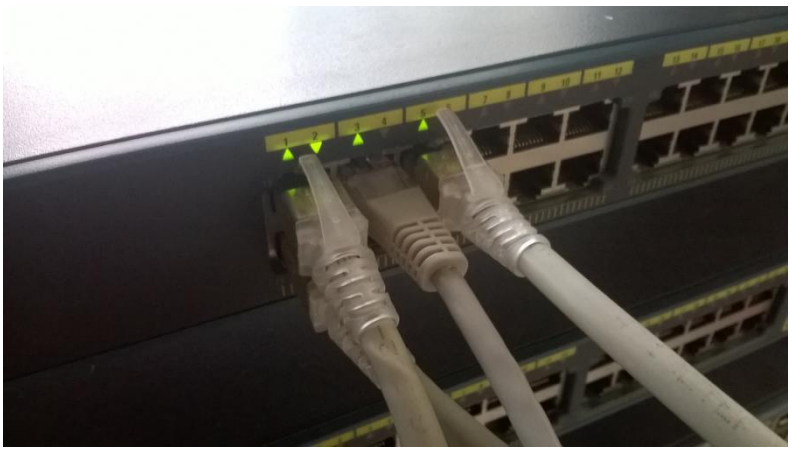

Figura 25. Conexión LAN a los Switch del Laboratorio, Fuente: **Autores del Proyecto**

Emulación en el Laboratorio de Redes y Telecomunicaciones

Se utilizó una red de topología real IPv6 Unicast Routing en el Laboratorio de Redes y Telecomunicaciones, en el cual como ya se mencionó antes cuenta con Un Servidor marca Hewlett Packard el cual se le tiene con el sistema operativo Ubuntu 16.04 LTS actualizado, Switch y Router de la marca CISCO (La Universidad cuenta con la certificación en CCNA) con soporte IPv6 y También con los computadores portátiles del laboratorio que cuentan con el sistema operativo Windows 1.

Figura 26. Laboratorio de Redes y Telecomunicaciones, Fuente: **UFPSO**

Figura 27. Sistema Operativo Virtual: HoneyDrive 3, Fuente: **Autores del Proyecto**

Sistema Operativo Virtual HoneyDrive 3

"HoneyDrive es la principal distribución de Linux de Honeypot. Es un dispositivo virtual (OVA) con la edición Xubuntu Desktop 12.04.4 LTS instalada. Contiene más de 10 paquetes de software Honeypot preinstalados y preconfigurados como el Honeypot Kippo SSH, **Dionaea** y Amun malware Honeypots, Honeyd Honeypot de baja interacción, Glastopf web Honeypot y Wordpot, Conpot SCADA / ICS Honeypot, Thug y PhoneyC Honeyclients y más. Además, incluye muchos scripts y utilidades preconfiguradas útiles para analizar, visualizar y procesar los datos que puede capturar, como Kippo-Graph, Honeyd-Viz, DionaeaFR, una pila ELK y mucho más. Por último, casi 90 herramientas conocidas de análisis de malware, análisis forense y monitoreo de red también están presentes en la distribución" (BruteForce lab, 2014)

Debido a todas las herramientas que tiene Honeydrive en cuanto a implementación de Honeypots, se eligió utilizarlo, en su versión 3 para la implementación del Honeypot 6Guard como sistema operativo esclavo del Ubuntu mediante Virtualbox.

Instalación de HoneyDrive por medio de VirtualBox

Se abrirá primero la pestaña **Archivo** y luego se seleccionara la opción **Importar Servicio Virtualizado.** Luego se buscara la ubicación del archivo OVA:

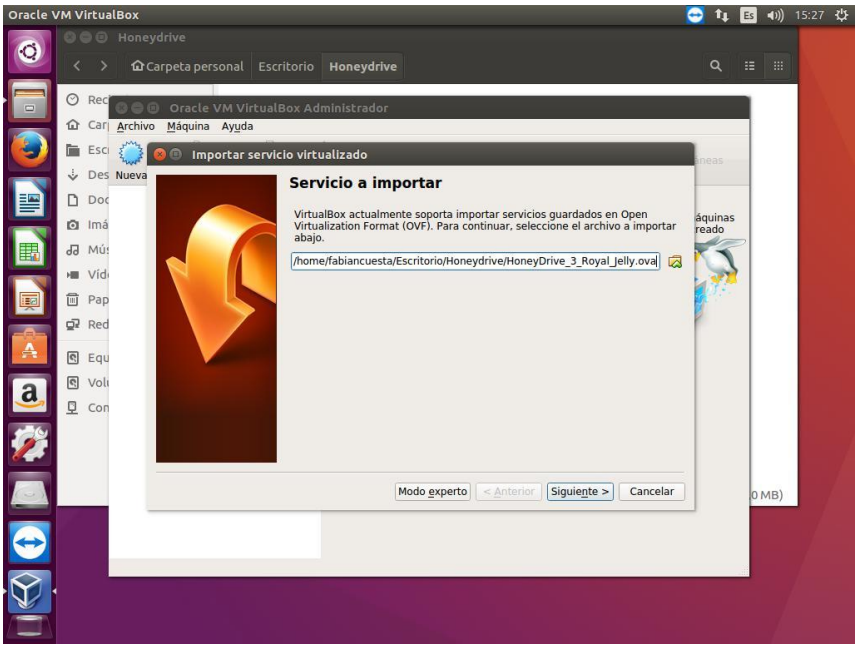

Figura 28. Instalación de HoneyDrive 3 por medio de VirtualBox, Fuente: **Autores del Proyecto**

Debido a que el servidor HP cuenta con dos puertos de red (**eno1 y eno2**), se utilizó el primer puerto para configurar su Red en el modo **Adaptador Puente**, el cual permite a la máquina virtual usar la tarjeta de red física del sistema operativo anfitrión para establecer su propia red (junto a la real) en la topología, esta será una **interfaz eth0** del Honeydrive que saldrá por **la eno1** del servidor HP con Ubuntu.

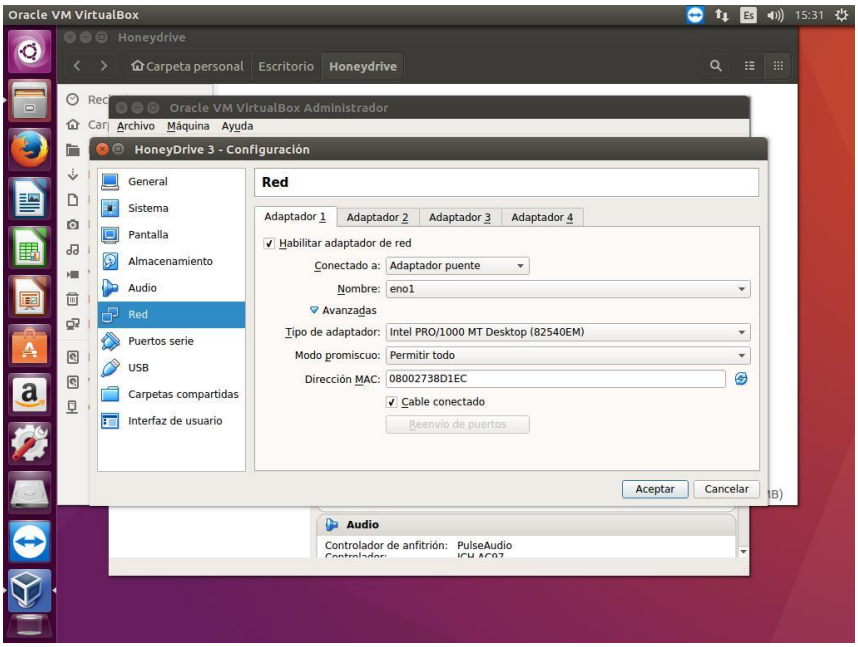

Figura 29. Configuración de Red Virtualbox para Honeydrive 3, Fuente: **Autores del Proyecto**

Y así nos da la bienvenida Honeydrive 3 con su pantalla principal de usuario. Su contraseña de usuario es **Honeydrive,** contiene un **Readme de texto** con instrucciones **para los Honeypots allí contenidos** y una Terminal renombrada a **"Terminator".** Pero aún no está listo para configurar los Honeypots:

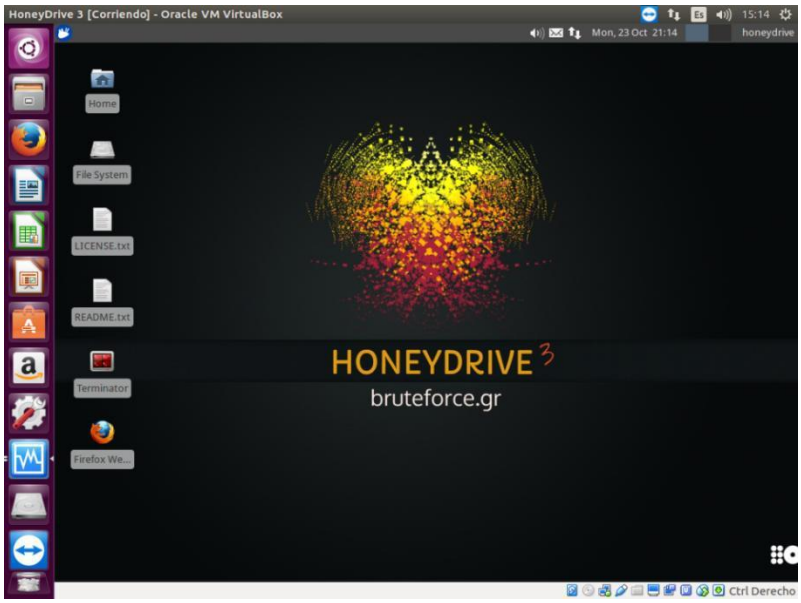

Figura 30. HoneyDrive 3 (corriendo), Fuente: **Autores del Proyecto**

Actualización de HoneyDrive 3

Al igual que Ubuntu es necesario actualizar Honeydrive 3 a Xubuntu para que funcione correctamente la implementación y configuración de elementos de la red Honeynet, Para actualizarlo realizaremos los pasos que se muestran a continuación:

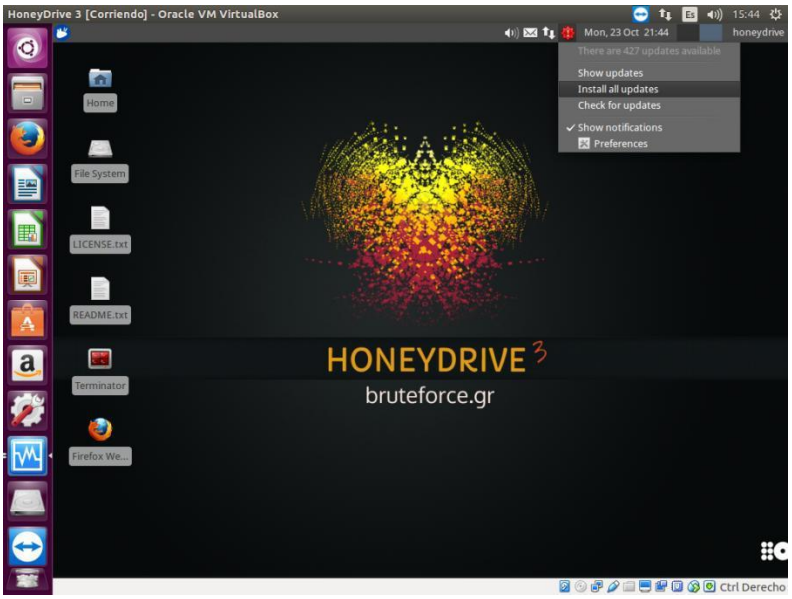

Figura 31. Actualización de HoneyDrive 3, Fuente: **Autores del Proyecto**

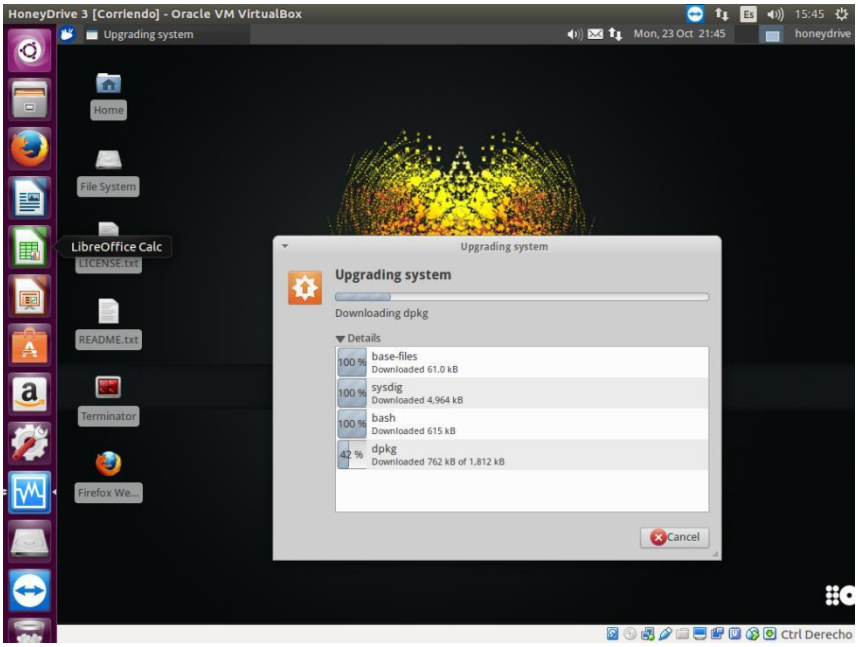

Figura 32. Ejecución de Actualizaciones de HoneyDrive 3, Fuente: **Autores del Proyecto**

Y luego después de reiniciar quedara listo nuestro sistema operativo Honeydrive.

Configuración IPv6 de Dispositivos

No podemos configurar aun el Honeypot, primero se debe realizar la Topología LAN IPv6 Unicast Routing que se muestra a continuación:

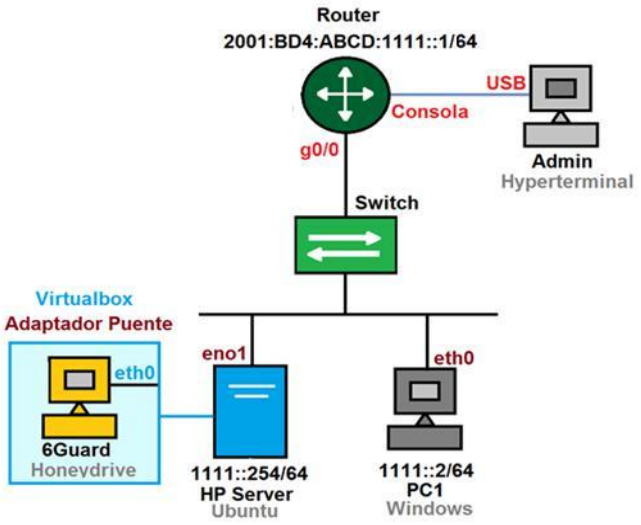

Figura 33. Topología LAN Honeynet IPv6 de Prueba Honeypots Fuente: **Autores del Proyecto**

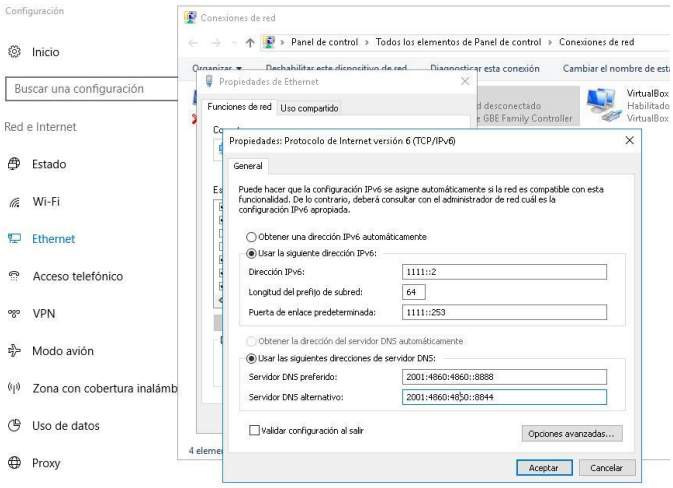

Figura 34. Configuración IPv6 del PC1 (Victima), Fuente: **Autores del Proyecto**

Configuración IPv6 del Router, su **interfaz g0/0**, por medio de un Computador que por medio del cable de **Consola** y por el software **Hyperterminal** realiza las configuraciones del Router.

```
Router>
Router>
Router>
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 uni
Router(config)#ipv6 unicast-routing
Router(config)#int g0/0
Router(config-if)#ipv6 address 2001:bd4:abcd:1111::1/64
Router(config-if)#no shu
Router(config-if)#no shutdown
Router(config-if)#_
```

Figura 35. Configuración IPv6 del Router, su **interfaz g0/0**, Fuente: **Autores del Proyecto**

Se pueden Utilizar los comandos ***show ip Interface brief*** y ***show running-config*** para verificar el estado de los puertos y su configuración respectivamente.

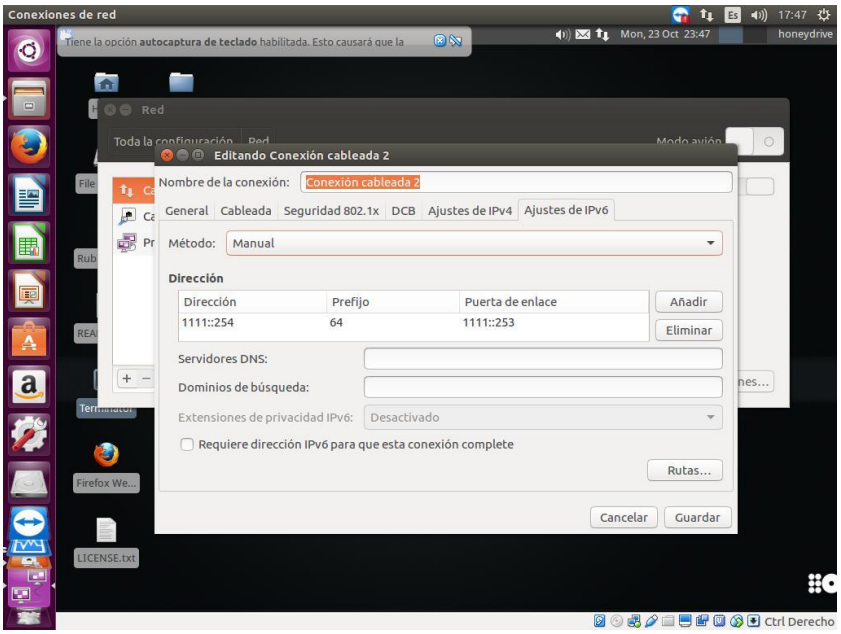

Figura 36. Configuración IPv6 del servidor Ubuntu (Interfaz **eno1**), Fuente: **Autores del Proyecto**

Configuración IPv6 automática de Honeydrive 3 (Interfaz **eth0, basada en la eno1**), visualizada mediante el comando ***ifconifg*** en la terminal.

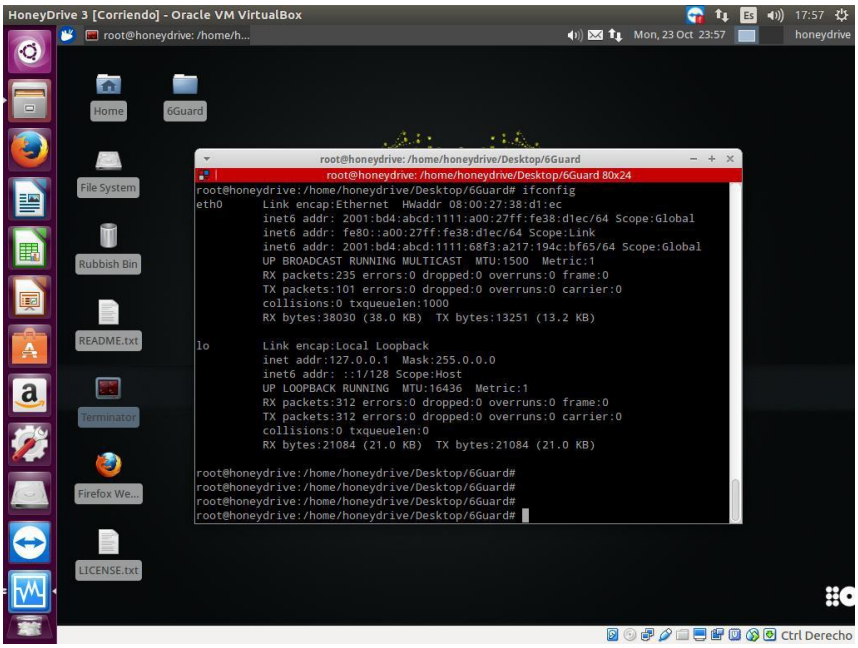

Figura 37. Configuración IPv6 automática de HoneyDrive 3, Fuente: **Autores del Proyecto**

Una vez hecho todo esto se procederá a instalar y configurar el Honeypot 6Guard.

6Guard Honeypot Detector de Ataques de la THC IPv6.

6Guard es un Honeypot especializado de baja interacción destinado a detectar ataques en la capa de red del modelo ISO / OSI, más específicamente ataques IPv6 de enlace local. Estos ataques se realizan sólo en un solo segmento de red separado por un Router. La ubicación de Honeypot óptima se conecta a un puerto de espejo de un Switch, pero se puede **conectar virtualmente** en cualquier lugar en LAN. Sin embargo, la eficacia de la detección de ciertos ataques es mucho mayor cuando se utiliza el puerto de espejo. (Sochor & Zuzcak, 2015)

El Honeypot 6Guard puede informar a su usuario sobre el ataque existente de enlace local de IPv6 y / o registrar la información sobre él, pero no pretende impedir tales ataques.

Arquitectura de 6Guard

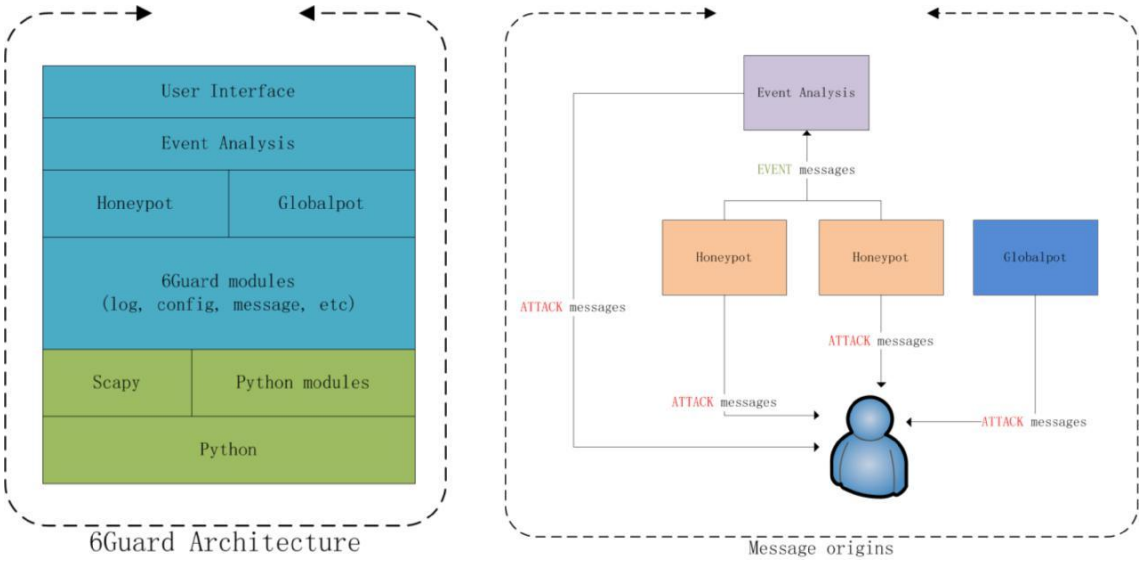

Figura 38. Arquitectura de 6Guard, Fuente: **The Honeynet Project**

6Guard se basa en Python y Scapy. Básicamente, contiene los tres módulos: **Honeypot**, **Globalpot** y **el de análisis de eventos**. (The Honeynet Project, 2012)

El **Honeypot** es un host virtual de baja interacción IPv6 con la capacidad de configuración automática de direcciones NDP y Stateless. En consecuencia, es responsable de detectar los ataques de **Unicast**. (The Honeynet Project, 2012)

El **Globalpot** es un módulo que se enfoca en detectar los ataques de **Multicast**. Dado que cada Honeypot sería capaz de capturar los ataques de Multicast al mismo tiempo, la implementación de la función como un punto global simplificaría la detección de manera significativa. (The Honeynet Project, 2012)

El módulo de análisis de eventos es responsable de analizar los mensajes de eventos y generar un mensaje de ataque si se detecta. Tal mecanismo es útil para detectar ataques como dos-new-ip6: cuando un Honeypot informaba un mensaje de Evento de que la dirección estaba en

uso, podría ser cierto, pero si más Honeypots informaban el mensaje de dirección en uso, podríamos saber que la red sufría un ataque dos-new-ip6. (The Honeynet Project, 2012)

Instalación y Configuración de 6Guard

El 6Guard no viene precargado en Honeydrive, sin embargo se puede descargar gratuitamente de la página de The Honeynet Project por medio de su **v1.0 tarball** o bien de su **Repositorio en Github**, e instalarlo y configurarlo perfectamente en el Honeydrive. Se descargó y se guardó en el escritorio del Honeydrive, su ruta quedó **home/honeydrive/Desktop/6Guard**

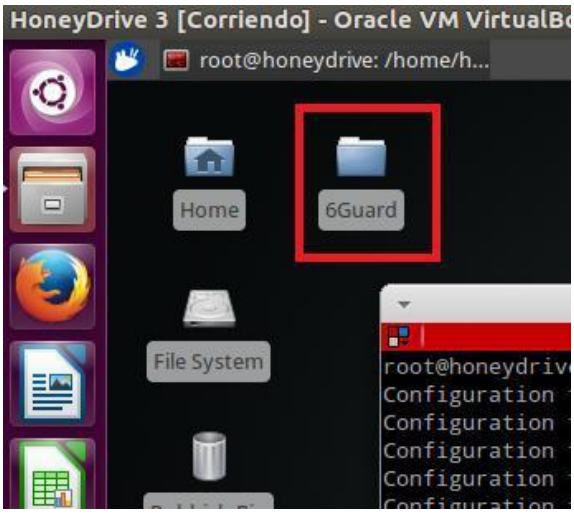

Figura 39. Carpeta de Instalación de 6Guard, Fuente: **Autores del Proyecto**

Lo primero es ejecutar la terminal en Honeydrive, y entrar como **Root** con el comando ***sudo su*** (Contraseña ***honeydrive***), aunque en las actualizaciones que realizamos de Honeydrive ya venga incluido el Scapy y Python, se confirmara que ya está instalado (o instalarlo si no lo está) con el comando ***apt-get install python-scapy***

Entraremos al directorio de ruta del 6Guard por medio del comando ***cd,*** y ejecutamos el comando ***sudo ./conf_generator.py*** para generar los archivos de configuración:

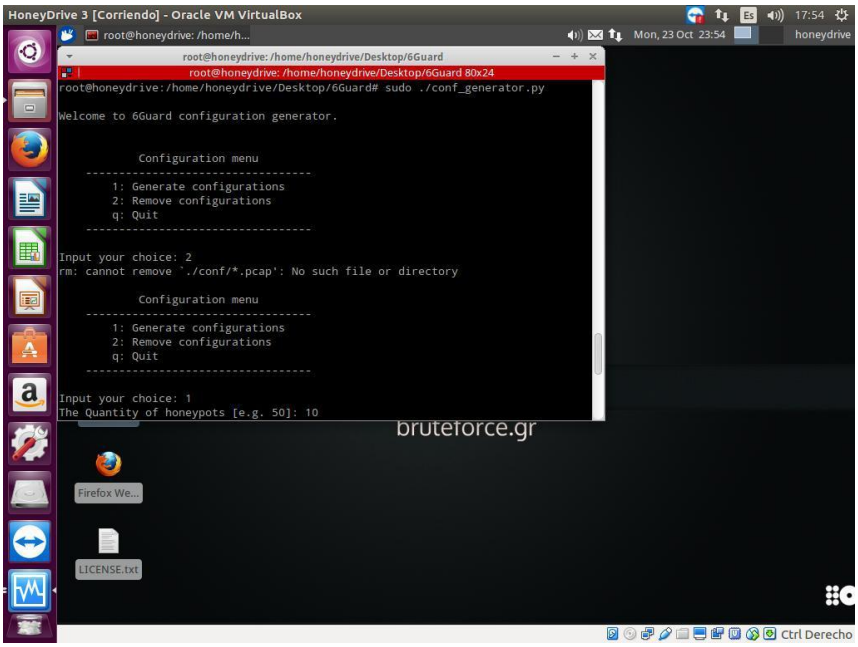

Figura 40. Menú de Configuración de 6Guard, Fuente: **Autores del Proyecto**

Se nos muestra un menú. **1 para generar las configuraciones, 2 para borrarlas** y **q para salir. Presionaremos 1.**

Luego se llenan los campos como se muestra la siguiente imagen, teniendo en cuenta la existencia del **Router Cisco,** la interfaz virtual **eth0** ya configurada previamente en **Virtualbox** y que la asignación de direcciones IPv6 **fue realizada**.

Presionamos **q** para salir y luego ejecutamos el comando ***sudo ./6guard.py*** para ejecutar el Honeypot 6Guard, en este caso **10** de ellos.

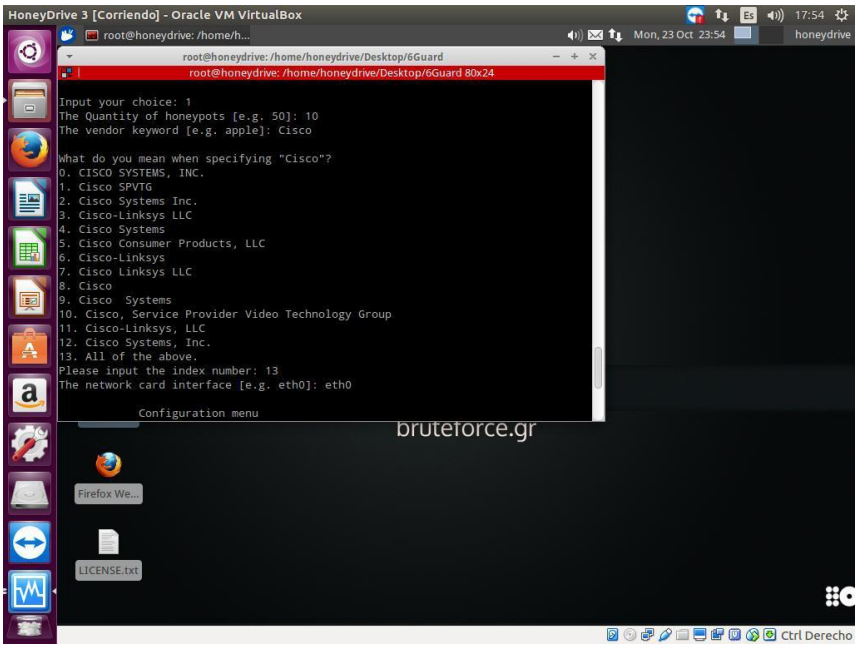

Figura 41. Especificaciones de 6Guard, Fuente: **Autores del Proyecto**

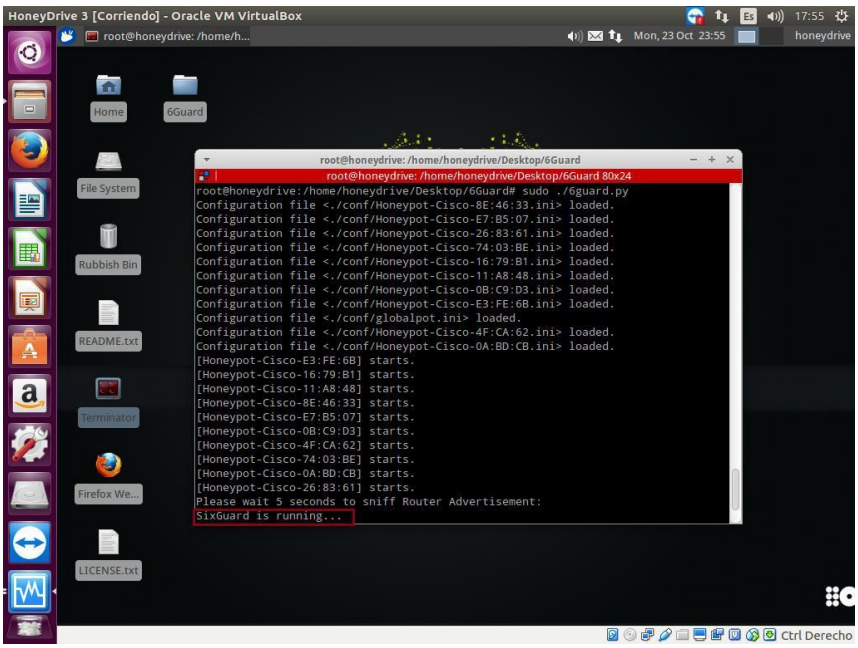

Figura 42. Ejecución de 6Guard, Fuente: **Autores del Proyecto**

Como es la primera vez que el 6Guard establece conexión externa con el Router Cisco, al momento de la ejecución del Honeypot, este mostrara la información de dicho Router, confirmándose su veracidad de conexión por medio de datos propios del Router Cisco como la dirección IPv6 que previamente se le asigno y su dirección Mac de origen (Source Link-Layer Address) y cumpliendo así con el objetivo de esta prueba.

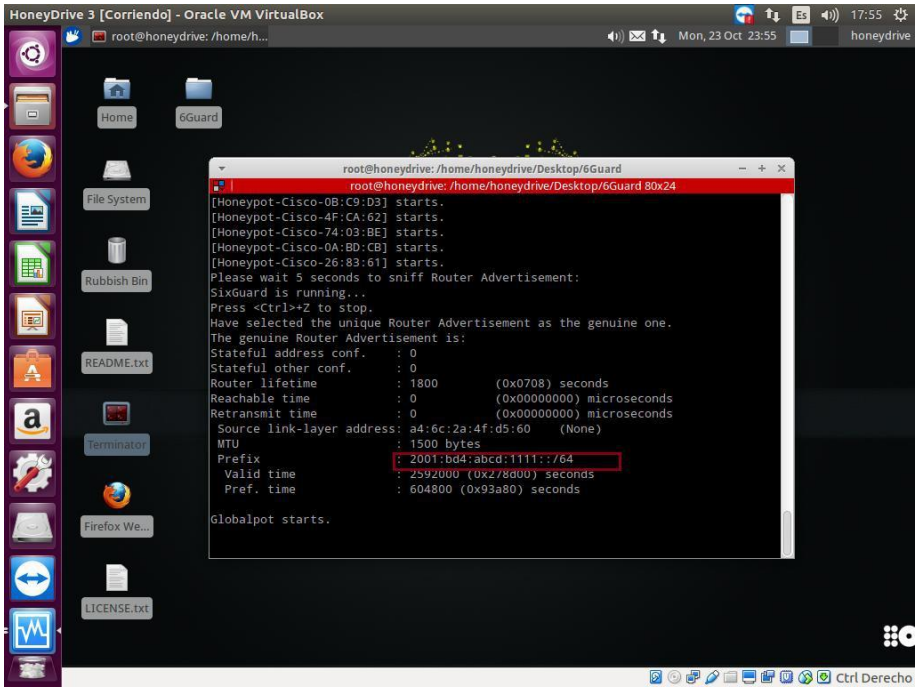

Figura 43. 6Guard: Reconocimiento del Router, Fuente: **Autores del Proyecto**

6Guard establece conexión externa con el Router Cisco Dentro de la carpeta de 6Guard están tres carpetas o directorios importantes:

El directorio. /conf almacena los archivos de configuración de Honeypots y Globalpot.

El directorio. /log almacena los registros de operación y los registros de ataque.

El directorio. /pcap almacena los paquetes relacionados con mensajes que se pueden revisar en **Wireshark**.

Resumiendo, cuando un atacante realice una acción ofensiva IPv6, en estos directorios quedara guardada esa información de ataque. Más adelante en la fase de implementación completa y la simulación de ataques, se mostrara a detalle esto.

DionaeaFR Con Soporte IPv6

Dionaea es uno de los Honeypots más populares y efectivos en la captura de registros de malware en **Internet,** sin embargo en la topología realizada se colocara bajo un ambiente **Lan IPv6**, y se utilizara su versión Front End, otro Honeypot llamado **DionaeaFR** que funciona como extensión**,** pero antes de implementarlo se tendrán en cuenta las siguientes consideraciones:

- Tanto Dionaea como DionaeaFR vienen implementados y precargados ya en **Honeydrive**.

- Ambos Honeypots utilizan los mismos directorios de archivos de configuración, binarios y logs.

- Dionaea **si** tiene soporte para IPv6 pero DionaeaFR **no**.

- DionaeaFR se puede modificar para que soporte **IPv6.**

- **DionaeaFR-IPv6** es el nombre que se eligió para la versión modificada con IPv6.

- **DionaeaFR-IPv6** será un Honeypot secundario y de soporte adicional en la red, ya que **6Guard** es más efectivo y nativo para IPv6.

- **No** es necesario actualizar los archivos de Honeydrive pero si es recomendable.

- **No** se recomienda actualizar Honeydrive a versiones de **Ubuntu** posteriores como la 14.04 etc…

- Se necesitara un **navegador** que soporte direccionamiento IPv6 para acceder a la interfaz gráfica de **DionaeaFR-IPv6**. Se eligió **Chromium**, el cual puede ser instalado desde la **terminal.**

- **No** es necesario configurar dirección IPv6 en el PC (Windows) que emula la máquina virtual con Honeydrive, pero **si** lo es configurar direccionamiento IPv6 para el Honeydrive, ya

que como este está en **modo adaptador puente**, formara parte de la topología de red como independiente:

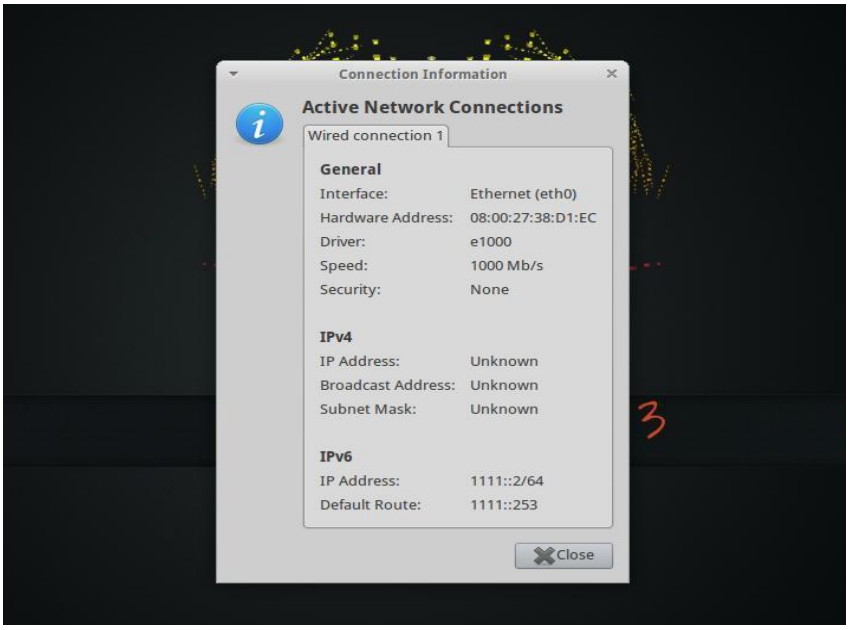

Figura 44. Información de Configuración IPv6 de HoneyDrive 3, Fuente: **Autores del Proyecto**

Conversión de DionaeaFR a DionaeaFR-IPv6

Como se mencionó anteriormente es necesario modificar ciertos archivos de DionaeaFR para qué soporte IPv6, la versión base IPv6 de DionaeaFR se puede descargar del repositorio **Github**: https://github.com/kevinvalk/DionaeaFR (Se le cambio el nombre de carpeta a "**DionaeaFR6"** y se guardó en el escritorio).

Es necesario modificar dos archivos: el primero es la extensión a la base de datos por medio del archivo modificado conections.html (https://github.com/rubenespadas/DionaeaFR/pull/12/files) de ruta **Desktop/DionaeaFR6/DionaeaFR/Templates/graphs/connections.html**

El segundo es el que contiene las configuraciones de la base de datos, el archivo **settings.py.dist (Desktop/DIonaeaFR6/DionaeaFR), este se modificara de la siguiente forma:**

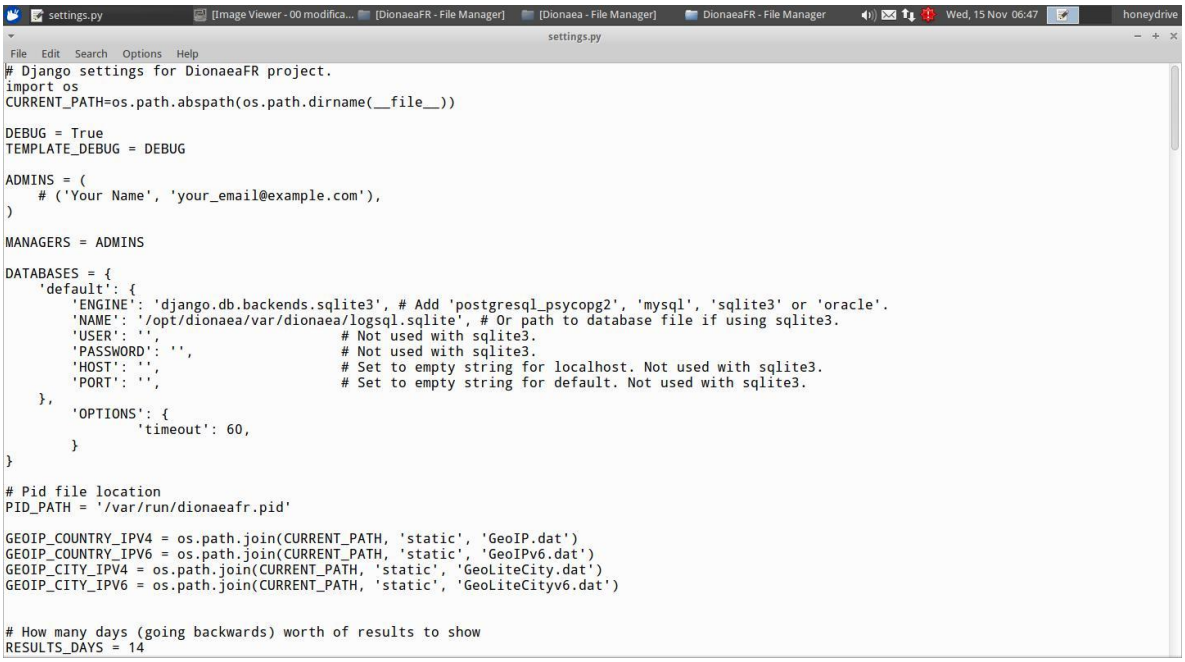

Figura 45. Conversión de DionaeaFR a DionaeaFR-IPv6, Fuente: **Autores del Proyecto**

Luego se guardara como **settings.py**

Siguiendo las instrucciones del archivo **Readme** se cargaran los archivos necesarios para ejecutar el Honeypot **DionaeaFR-IPv6,** sobre todo los de **Direccionamiento IPv6 Geográfico,** los cuales se descargarán, se descomprimirán y se moverán a la carpeta del Honeypot en cuestión. Estos aportan información geográfica de ataques por medio de un mapa en la interfaz:

-Información de ataques IPv4: **GeoIP.dat** y **GeoLiteCIty.dat** (Originales de **DionaeaFR)**

-Información de ataques IPv6: **GeoIPv6.dat** y **GeoLiteCItyv6.dat** (Añadidos para **DionaeaFR-IPv6)**

Se abre **Terminator,** se ingresa al directorio de **DionaeaFR-IPv6 (Desktop/DionaeaFR6),** para luego ejecutar el comando ***python manage.py collectstatic*** el cual se encargara de mover los archivos originales de la carpeta **static** del Honeypot **DionaeaFR** original precargado en

Honeydrive, y los archivos de **Direccionamiento IPv6 Geográfico**, a la carpeta **static** del Honeypot **DionaeaFR-IPv6 (Desktop/DionaeaFR6/DionaeaFR).** Este comando es vital para el correcto funcionamiento del Honeypot con soporte **IPv6**:

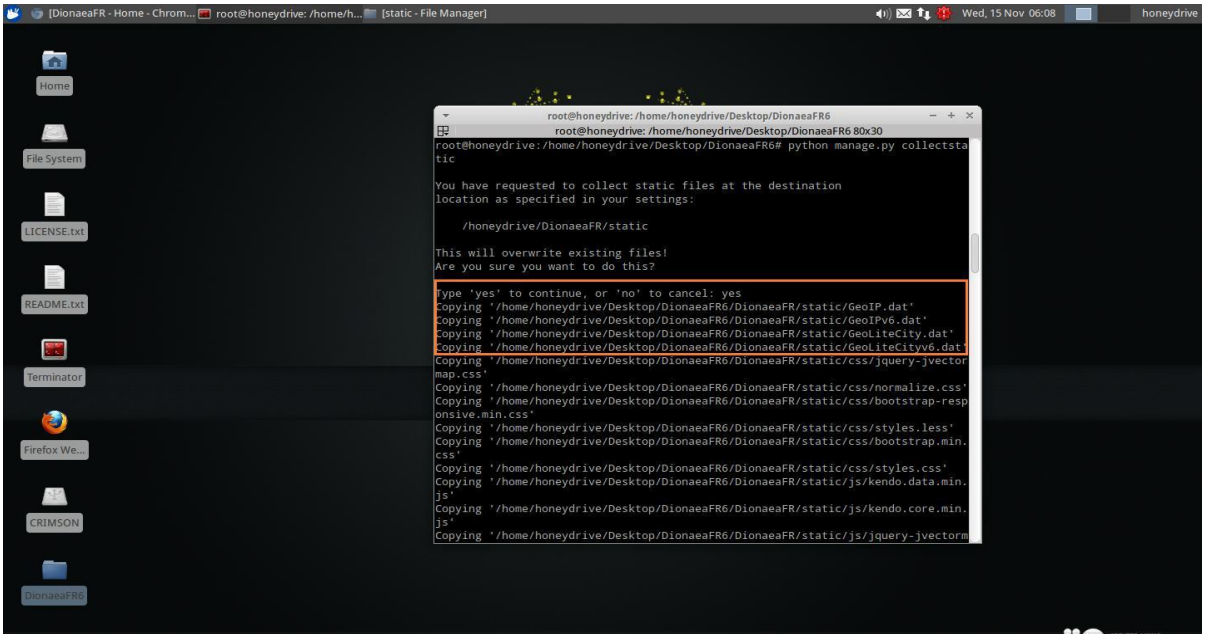

Figura 46. Instalación de Archivos de Direccionamiento IPv6 Geográfico, Fuente: **Autores del Proyecto**

A continuación, se revisa la Carpeta **static** del Honeypot **DionaeaFR-IPv6,** los archivos que son necesarios para la configuración de la interfaz gráfica, y los archivos **.dat**:

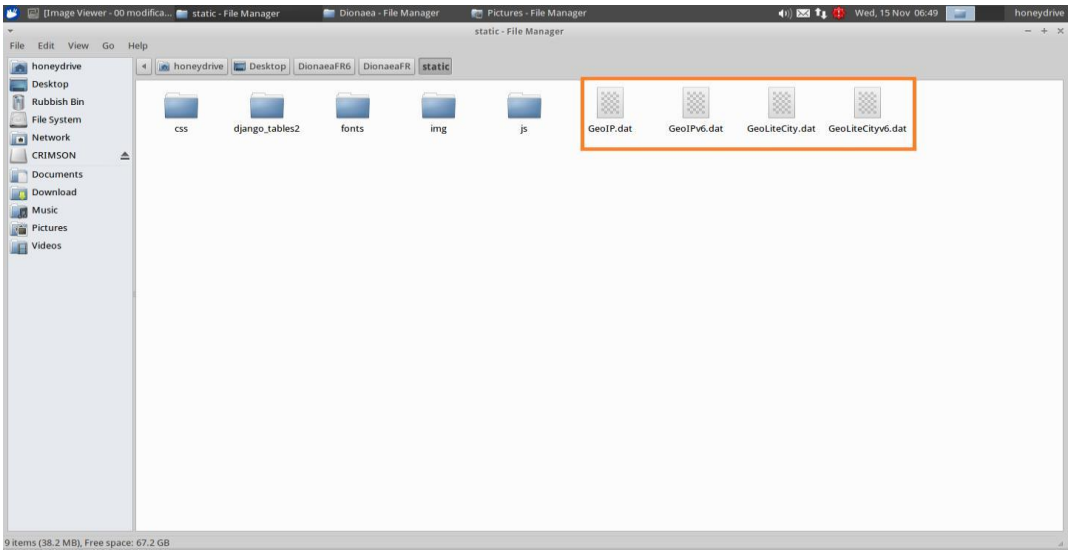

Figura 47. Carpeta static del Honeypot DionaeaFR-IPv6, Fuente: **Autores del Proyecto**

Ejecución de DionaeaFR-IPv6

Una vez realizados todos los pasos anteriores, se procede a ejecutar el Honeypot modificado, mediante el siguiente comando: ***python manage.py runserver [dirección-IPv6 activa]:puerto.*** Si todo fue realizado correctamente, el Honeypot **DionaeaFR-IPv6, dirá que no hay errores en su validación y quedara a la escucha de ataques:**

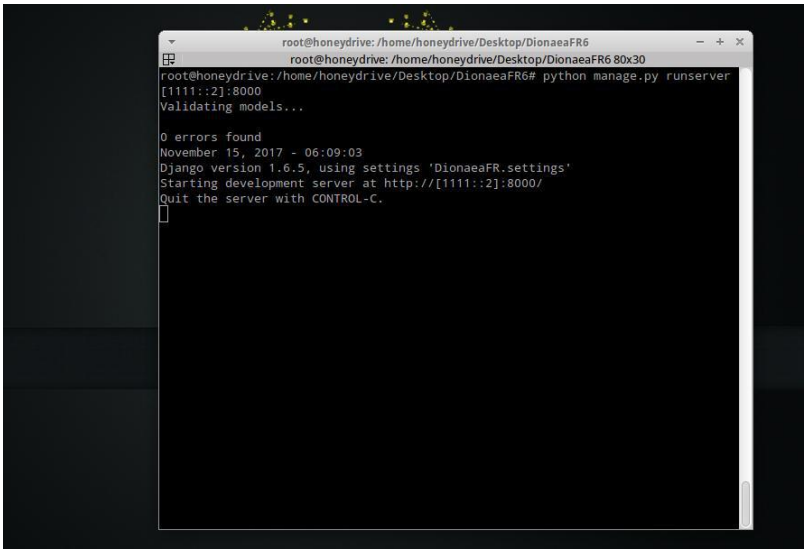

Figura 48. Ejecución de DionaeaFR-IPv6, Fuente: **Autores del Proyecto**

En la fase de pruebas de ataques más adelante, se mostrara con detalle cómo se registran los ataques detectados por **DionaeaFR-IPv6** en la **terminal**.

También es necesario ejecutar el **Dionaea** en otra terminal mientras DionaeaFR-IPv6 se ejecuta con el comando ***/opt/dionaea/bin/dionaea***

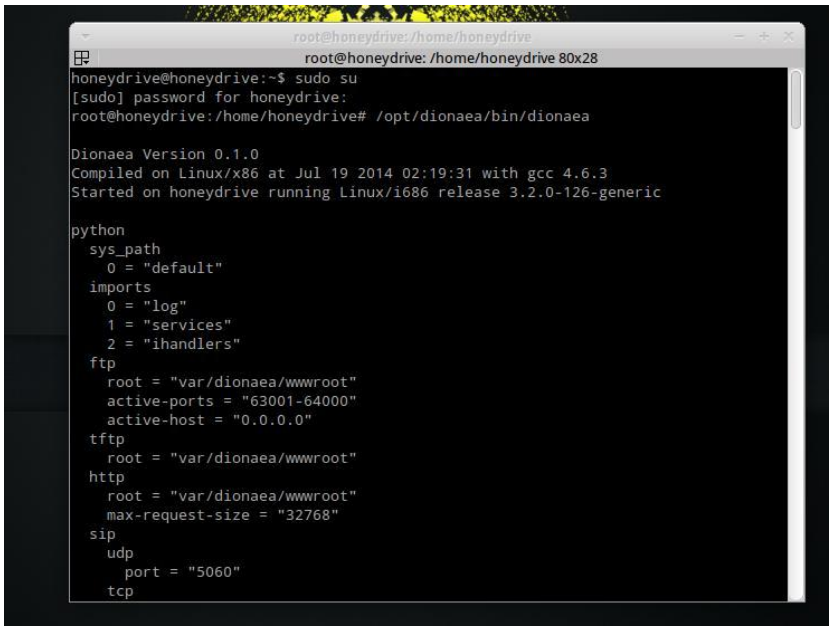

Figura 49. Ejecución de Dionaea, Fuente: **Autores del Proyecto**

Mientras **Dionaea** con su framework **DionaeaFR-IPv6** está a la escucha de ataques en la terminal, se entrara al navegador (en este caso **Chromium**) y se introducirá la dirección IPv6 más el puerto previamente configurado. De esta forma se accede a la interfaz gráfica de **DionaeaFR-IPv6:**

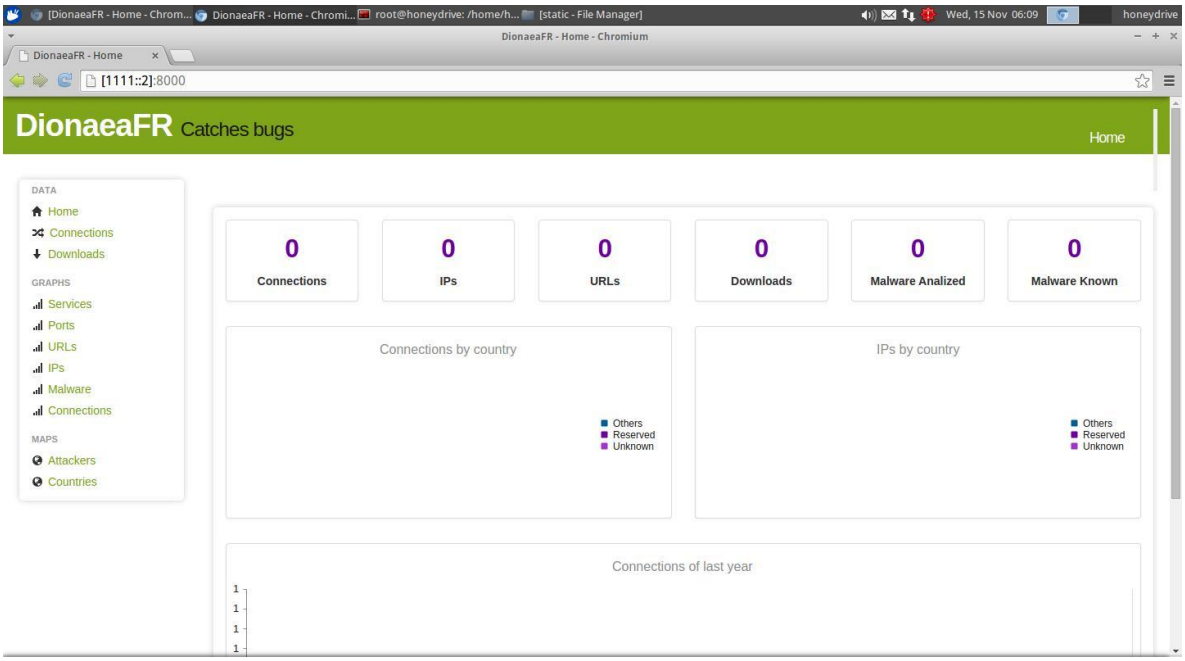

Figura 50. Interfaz Gráfica de DionaeaFR-IPv6, Fuente: **Autores del Proyecto**

Hay varias opciones que se pueden explorar en el menú de la parte izquierda junto a información general de ataques registrados en tiempo real. La opción de **Mapas** se encargara de confirmar que es efectivamente **DionaeaFR-IPv6** ya que tiene que reconocer y separar ataques IPv4 de ataques IPv6:

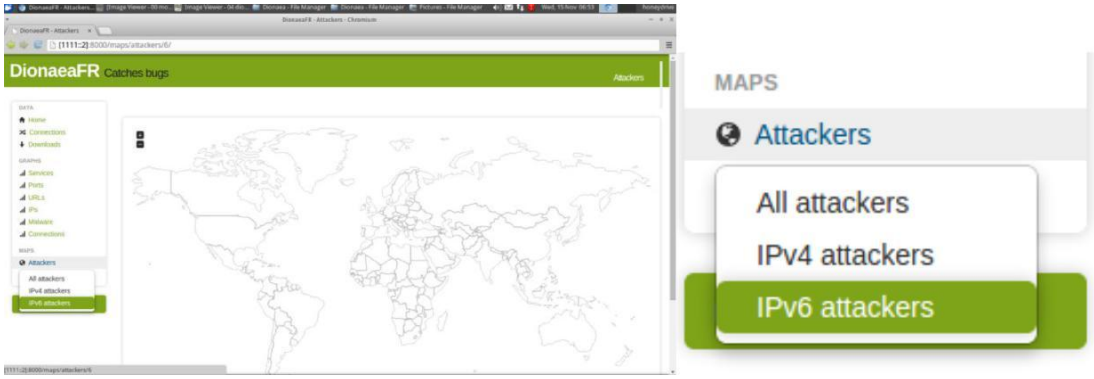

Figura 51. Mapa de Ataques IPv6, Fuente: **Autores del Proyecto**

Como Dionaea está enfocado a entornos de Internet, se deja en la Honeynet LAN en el caso de que esta se conecte a internet **DionaeaFR-IPv6** en su **Interfaz Gráfica**.

4.3 Evaluar el funcionamiento de la red Honeynet, analizando métodos y técnicas de atacantes obtenidos a través de esta, para plantear medidas de aseguramiento.

Incompatibilidad de Honeywall Gateway con IPv6

Como se ha aclarado anteriormente, el **Honeywall Gateway** es un elemento fundamental en una red Honeynet, brinda potencia ya que actúa como **La Gestión Unificada de Amenazas (UTM)** de la Honeynet, pero **sin Cortafuegos**, contiene muchas herramientas adaptadas (**Daemons**) y configuradas a él como **SNORT**; **su pieza central,** un **Sistema Detector de Intrusos (IDS)** bastante eficiente. **Sebek**, un Sniffer de red, definido como "una herramienta de captura de datos diseñada para registrar las actividades del atacante en un Honeypot, sin que el atacante (con suerte) lo sepa." (The Honeynet Project, 2006). Es el elemento de la Honeynet que todo lo ve (De allí el nombre de su interfaz web **Walleye**), todo lo que suceda en dicha red será visto y registrado por el Honeywall Gateway.

CDROM Honeywall Roo v1.4

El **CDROM Honeywall Roo v1.4** proporcionado por **The Honeynet Project,** es una colección de varios software de Código Abierto (OpenSource), que incorpora el Honeywall Gateway Mediante una Instalación guiada y una posterior configuración Manual. **Su archivo de imagen .ISO puede ser descargado gratuitamente en la página de The Honeynet Project**.

"Honeywall CDROM es un CDROM de arranque que instala todas las herramientas y funcionalidades necesarias para crear rápidamente, mantener fácilmente y analizar efectivamente una **Honeynet de tercera generación**; Es el sucesor de **CDROM Eeyore.** En muchos sentidos, el original CDROM Eeyore fue un prototipo, para demostrar las capacidades de una Honeynet independiente, y aprender del CDROM. El nuevo CDROM Roo es diferente. Esto se considera

una solución de producción. Es más fácil de instalar y mantener y se puede implementar en grandes cantidades. Nuestro objetivo es que Honeynet salga del mundo de la investigación académica y se expanda como una solución real para una variedad de organizaciones.” (The Honeynet Project, 2006)

CDROM Honeywall Roo es el elemento que diferencia a una Honeynet de tercera generación de una de segunda. Aunque ambas tengan la misma arquitectura sus diferencias son las mejoras en el despliegue y la administración en Honeynets Gen III junto con la adición de Sebek integrado en la puerta de enlace. “Esto es lo que se conoce como Honeywal”.l (Abbasi F. , 2009)

Su Sistema Operativo de base inicial era Fedora pero a partir de su versión 1.3 fue cambiado a CentOS. “Se seleccionó CentOS porque no requirió muchos cambios en nuestro proceso de compilación y porque tiene un período de soporte más extenso.” (The Honeynet Project, 2007)

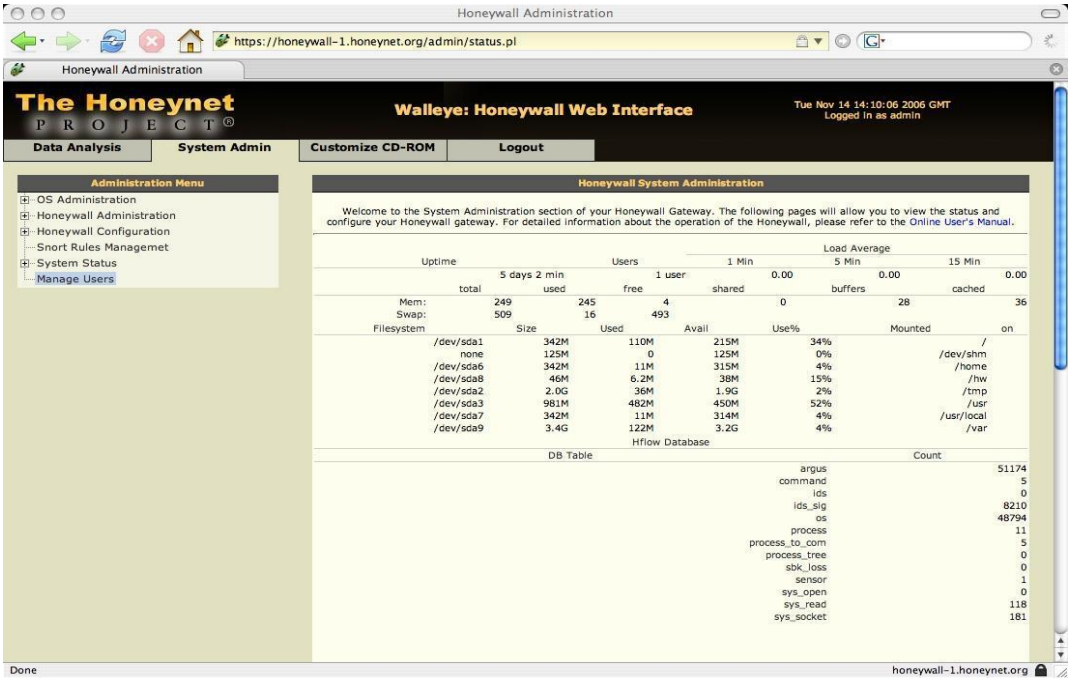

Figura 52. Walleye Interfaz web del Honeywall Roo 1.4, Fuente: **The Honeynet Project**

¿Honeywall Gateway IPv6?

"Honeywall CDROM es nuestra herramienta principal de alta interacción para capturar, controlar y analizar ataques. Crea una arquitectura que le permite desplegar Honeypots de baja interacción y alta interacción, **pero está diseñado principalmente para alta interacción**." (The Honeynet Project, 2007)

Como The Honeynet Project explica, el Honeywall CDROM está diseñado más que todo para Honeynets de alta interacción. Se hizo una revisión extensa de fuentes y una búsqueda profunda de documentación para una configuración de un **Honeywall Gateway en redes IPv6** (ya que en su manual no se especifica), pero no se encontró ninguna documentación correspondiente para esto. **Todos los documentos y configuraciones disponibles estaban orientados para IPv4,** a esto también cabe recordar que **los estándares de Honeypots para IPv6 están diseñados todos para baja interacción.**

Al estar el interrogante si un **Honeywall Gateway pudiera ser implementado en una Honeynet IPv6**, se procedió a realizar la Instalación y configuración del **CDROM Honeywall Roo 1.4** en el servidor HP con la siguiente topología de red LAN Honeynet:

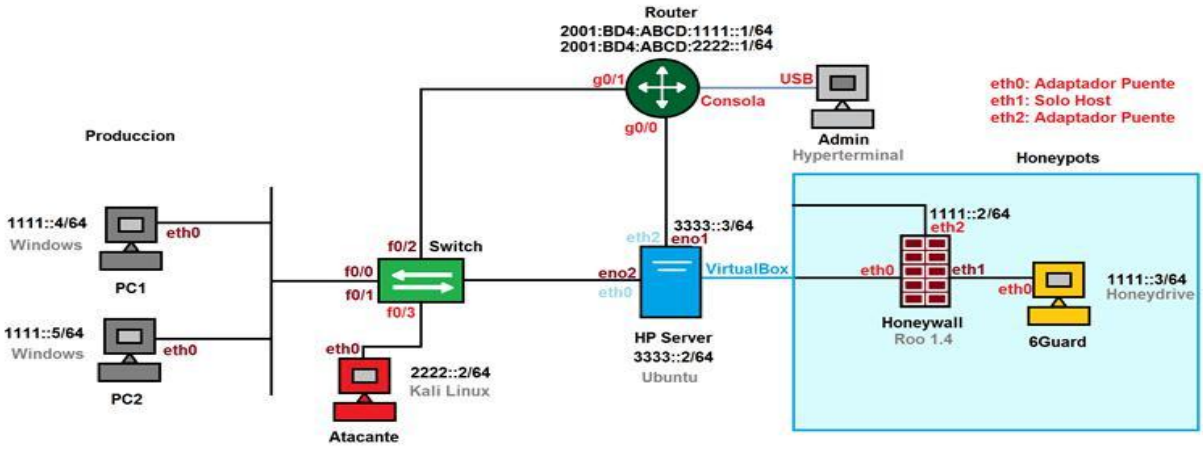

Figura 53. Topología LAN IPv6 de Prueba Honeywall Gateway, Fuente: **Autores del Proyecto**

La configuración de las interfaces del Honeywall es la siguiente:

-eth0 *configurada en* ***modo adaptador puente*** *para que salga a la red por medio de* ***eno2.*** *Esta conecta a los PCs físicos, victimas y atacantes, en una porción red de la Honeynet denominada "Producción".*

-eth1 *configurada en* ***modo solo-host (****también llamado* ***solo-anfitrión)*** *para que por medio de un adaptador virtual genérico (****vboxnet0)*** *se pueda conectar el* ***Honeypot Virtual****. Nota:* ***Varios Honeypots*** *en diferentes SOs invitados se pueden conectar por medio de un* ***Switch Virtual****.*

-eth2 *configurada en* ***modo adaptador puente*** *por medio de* ***eno1 directamente al Router,*** *para que en* ***uno o varios PC*** *(server HP) de la red se pueda acceder a la interfaz de configuración* ***Walleye****.*

Preparación De La Máquina Virtual.

Roo 1.4 al igual que **Honeydrive,** es una herramienta exclusiva para ser trabajada mediante un **ambiente de máquina virtual**, así que es necesario configurar apropiadamente **Virtualbox** para realizar la instalación correctamente.

Lo primero es crear un adaptador de red virtual en modo **Solo-Host,** este modo a diferencia del modo **Adaptador Puente**, crea una red interna a la que también pertenecerá el equipo Anfitrión (Sever HP, Ubuntu 16.04 LTS), esto para poder realizar las configuraciones multi Ethernet que requiere el Roo para funcionar, **necesita de tres interfaces**. **Como el servidor HP tiene dos interfaces físicas (eno1 y eno2) se procederá a crear un solo adaptador virtual para la tercera interfaz.**

Se presiona las teclas **ctrl+g** para entrar a las preferencias del Virtualbox, en la opción **Red**, se selecciona la **pestaña Redes solo-anfitrión**, para añadir un nuevo adaptador de red virtual.

Este se llama por defecto **vboxnet0** y su **configuración** es automática pero se puede modificar manualmente.

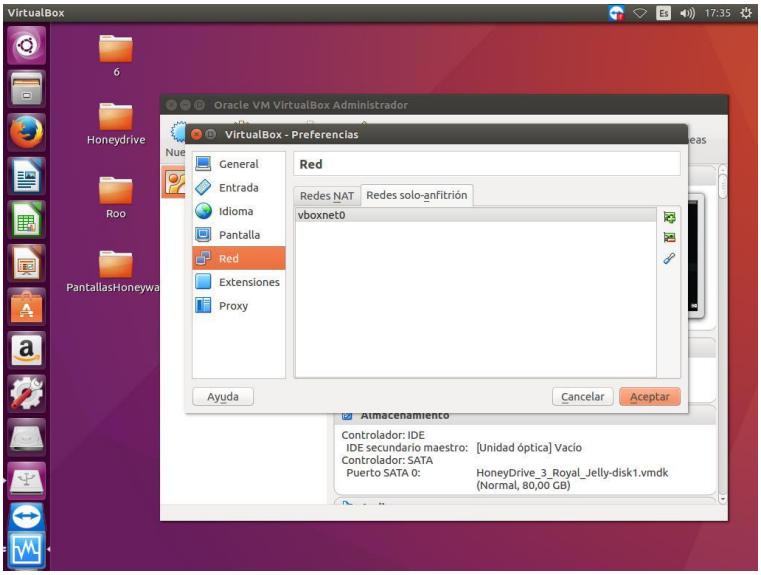

Figura 54. Configuración red Virtualbox, Modo Solo-Host, Fuente: **Autores del Proyecto**

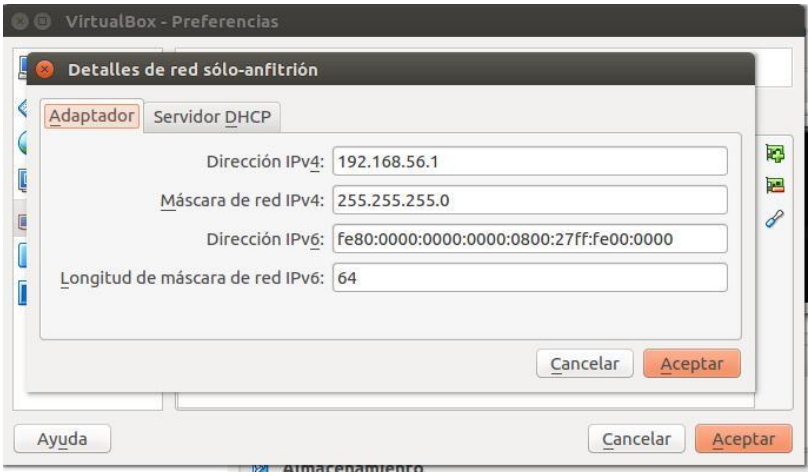

Figura 55. Configuración automática de vboxnet0, Fuente: **Autores del Proyecto**

De esta forma se tienen las siguientes tres interfaces:

I) eno2 configurada en la máquina virtual en modo adaptador puente como **eth0.**

Configuración de eno2 en Ubuntu:

Figura 56. Configuración de eno2 en Ubuntu, Fuente: **Autores del Proyecto**

Configuración de eth0 en Virtualbox:

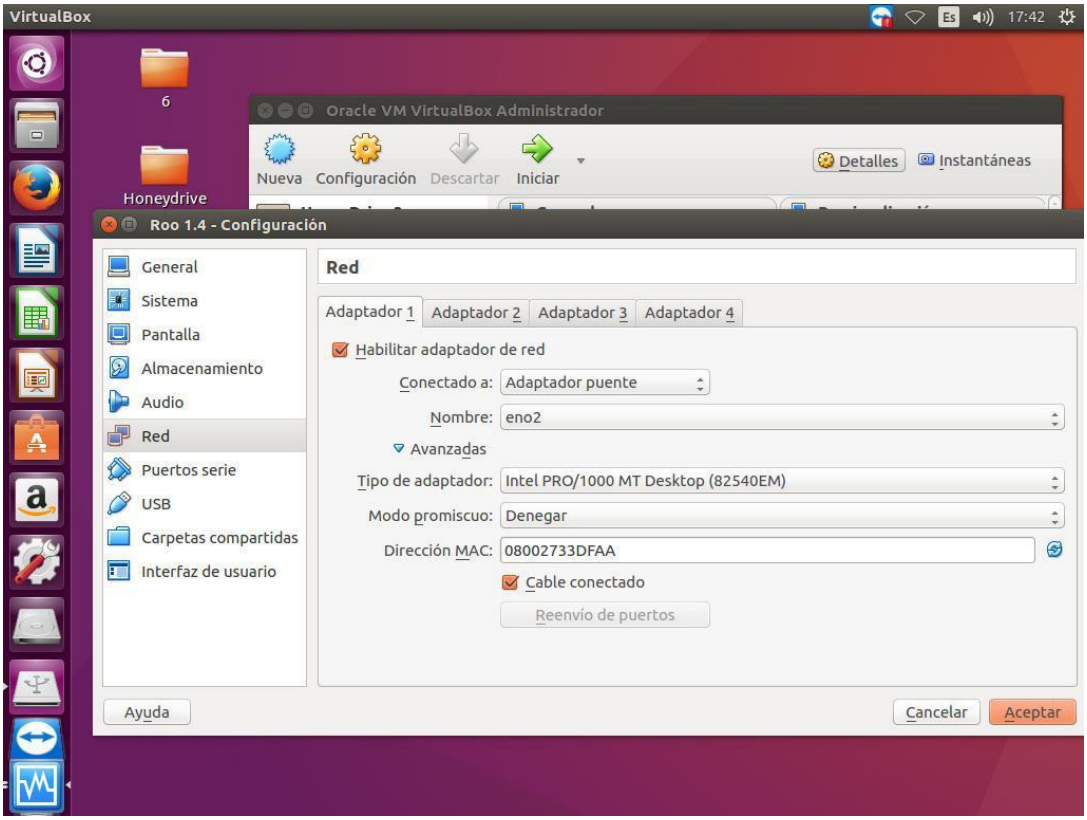

Figura 57. Configuración de eth0 en Virtualbox, Fuente: **Autores del Proyecto**

II) vboxnet0 configurada en la máquina virtual en modo adaptador solo host como **eth1.**

Configuración de vboxnet0 en Ubuntu:

Figura 58. Configuración de vboxnet0 en Ubuntu, Fuente: **Autores del Proyecto**

Configuración de eth1 en Virtualbox:

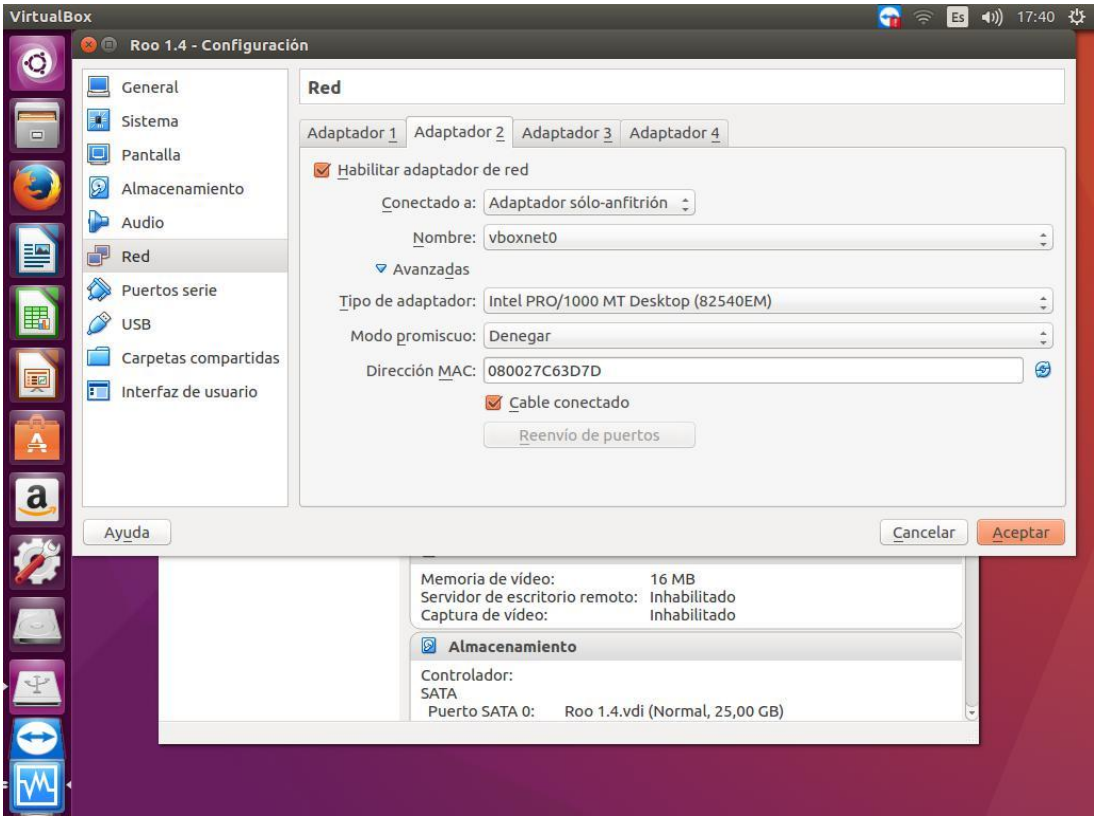

Figura 59. Configuración de eth1 en Virtualbox, Fuente: **Autores del Proyecto**

III) eno1 configurada en la máquina virtual en modo adaptador puente como **eth2.**

Configuración de eno1 en Ubuntu:

Figura 60. Configuración de eno1 en Ubuntu, Fuente: **Autores del Proyecto**

Configuración de eth2 en Virtualbox:

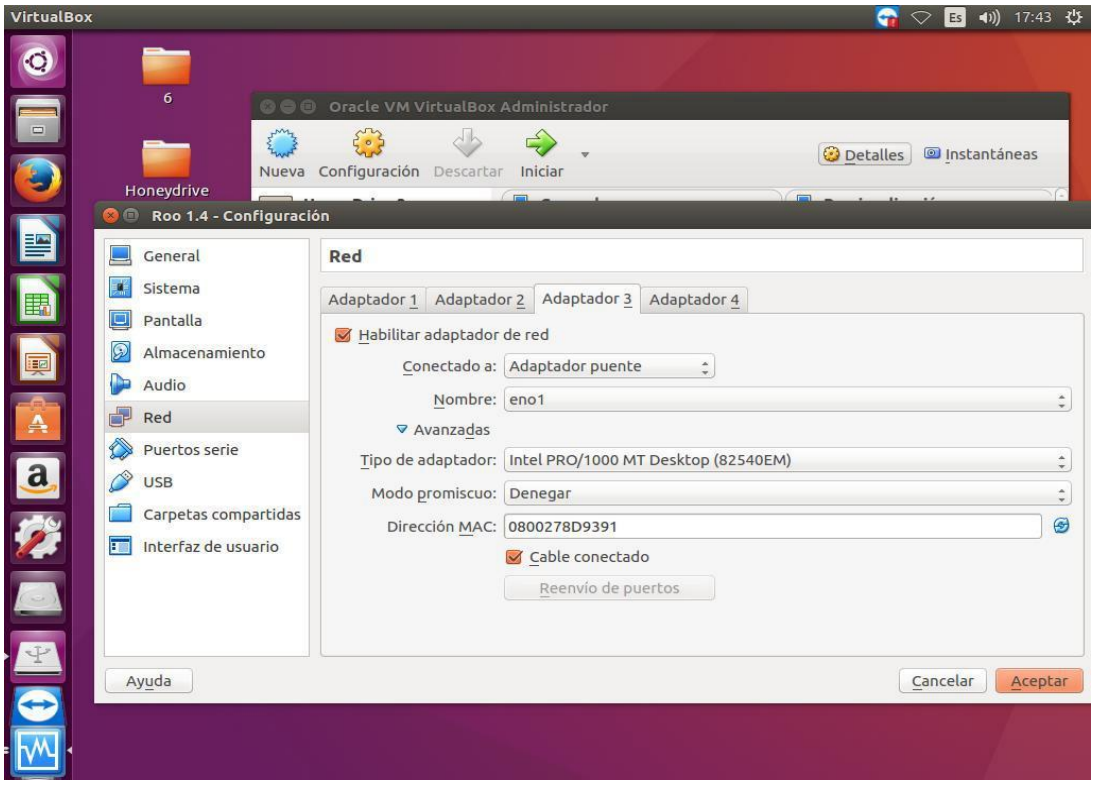

Figura 61. Configuración de eth2 en Virtualbox, Fuente: **Autores del Proyecto**

A continuación se procede a montar la imagen .ISO de Roo 1.4 en Virtualbox. La Red y la configuración de Virtualbox están listas para proceder a la instalación y configuración del CDROM Honeywall Roo 1.4:

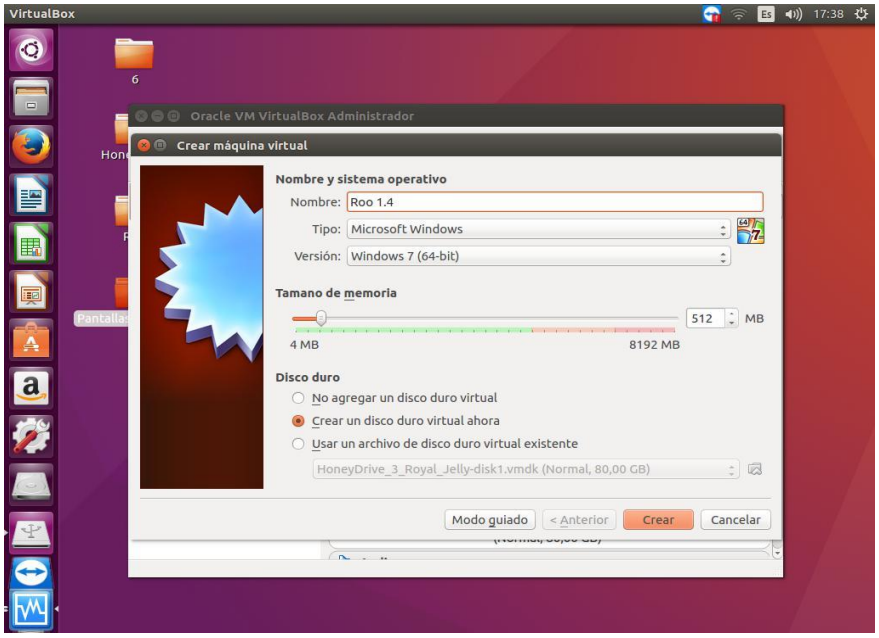

Figura 62. Creación de Máquina Virtual para Roo 1.4, Fuente: **Autores del Proyecto**

Se selecciona la imagen .ISO de Roo 1.4 para ser cargada en la unidad de cd virtual mediante la opción **almacenamiento** del sistema operativo invitado:

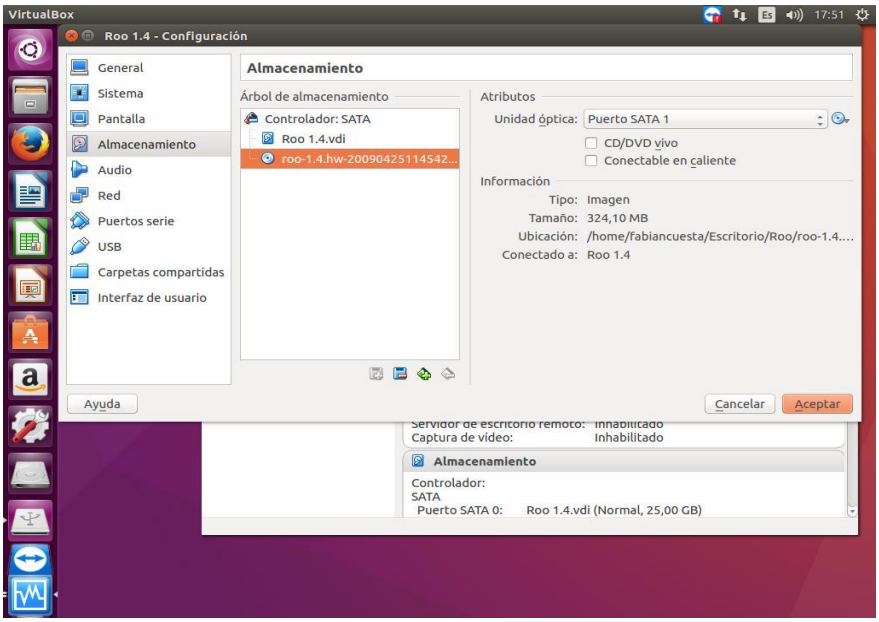

Figura 63. Carga de la imagen .ISO de Roo 1.4, Fuente: **Autores del Proyecto**

Instalación y Configuración de Honeywall Roo.

Cuando ya está todo correctamente configurado (**eth0, eth1 y eth2**), se procede a realizar la instalación del Honeywall Roo 1.4, se muestra la pantalla inicial, en la cual solo es necesario apretar una tecla para que comience a realizar la instalación

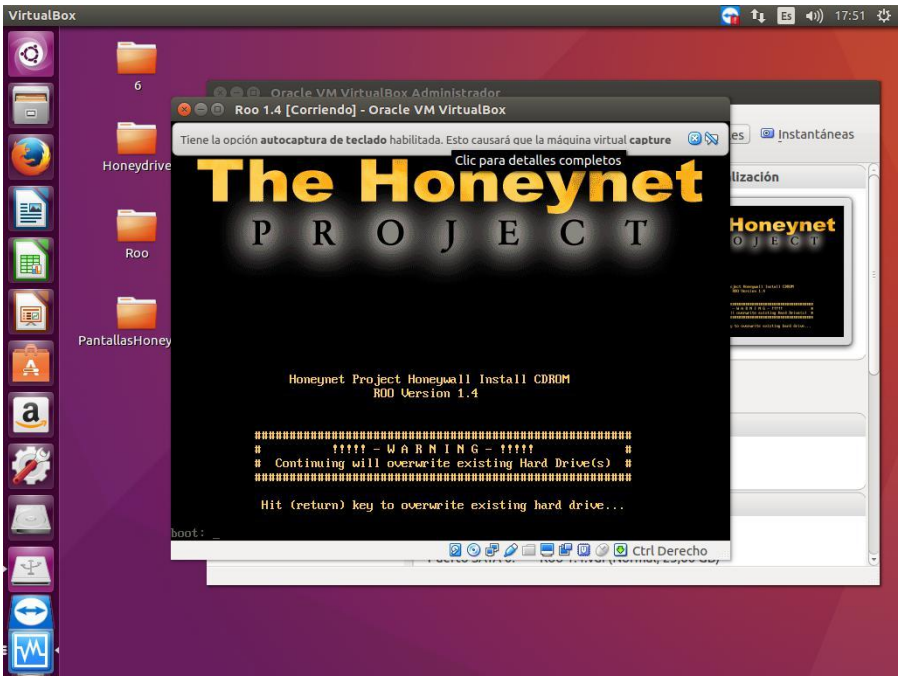

Figura 64. Pantalla Inicial de Roo 1.4, Fuente: **Autores del Proyecto**

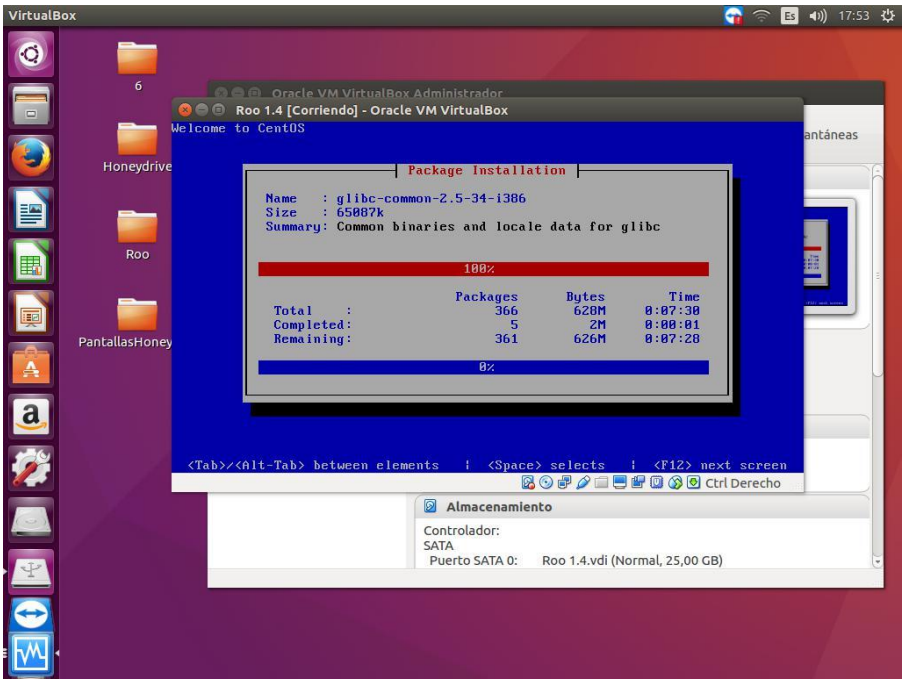

Figura 65. Proceso De Instalación de Roo 1.4, Fuente: **Autores del Proyecto**

Luego de terminar la instalación, Roo se reiniciara, cargara sus archivos y se pondrá en marcha una consola de comandos. Es necesario saber que hay dos usuarios en esta consola, uno es **roo (**Contraseña: **honey**), y el otro es el superusuario **root (**Contraseña: **honey**), al cual se necesita acceder por medio del comando ***su -*** .

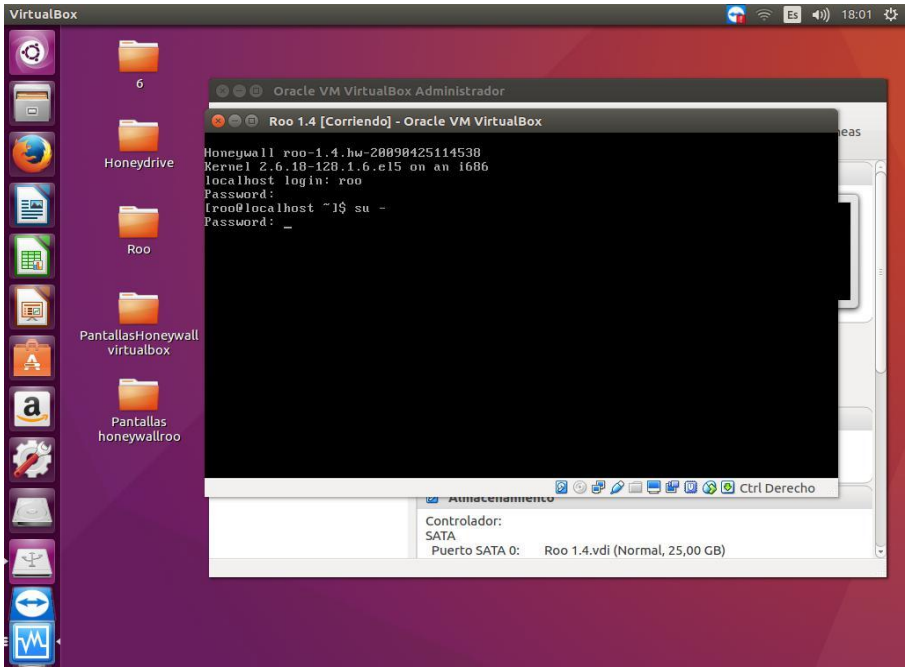

Figura 66. Super Usuario Root, Fuente: **Autores del Proyecto**

Inmediatamente después saldrá el menú de configuración de Roo 1.4

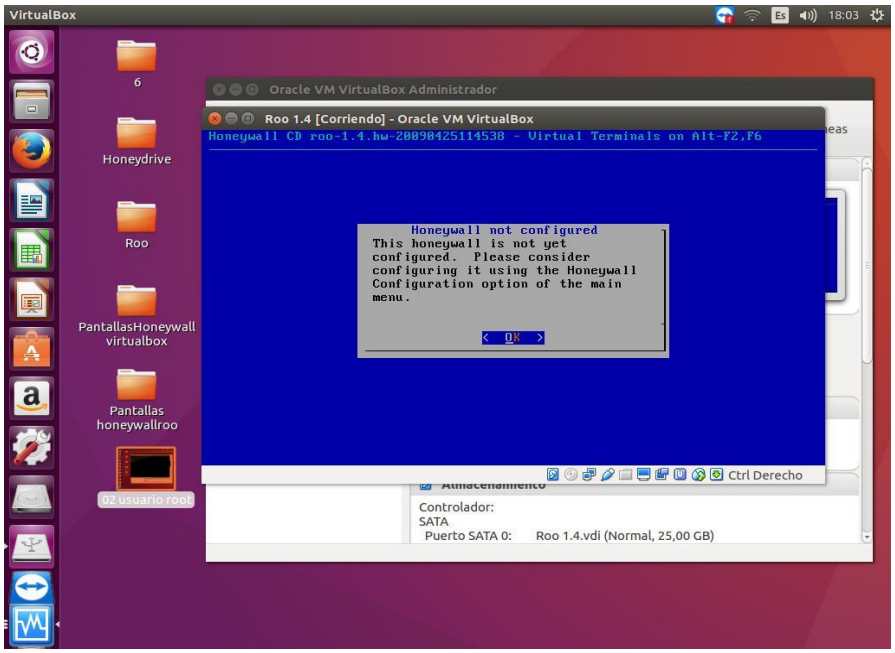

Figura 67. Mensaje de Entrada de Roo 1.4, Fuente: **Autores del Proyecto**

A partir de este momento se evidenciara las pantallas de las opciones más importantes durante el proceso de configuración de Roo 1.4 ya que es un poco extenso. En el menú principal se elegirá **la opción 4, Configuración de Honeywall y** luego el modo de configuración inicial será la **opción 3, Entrevista.**

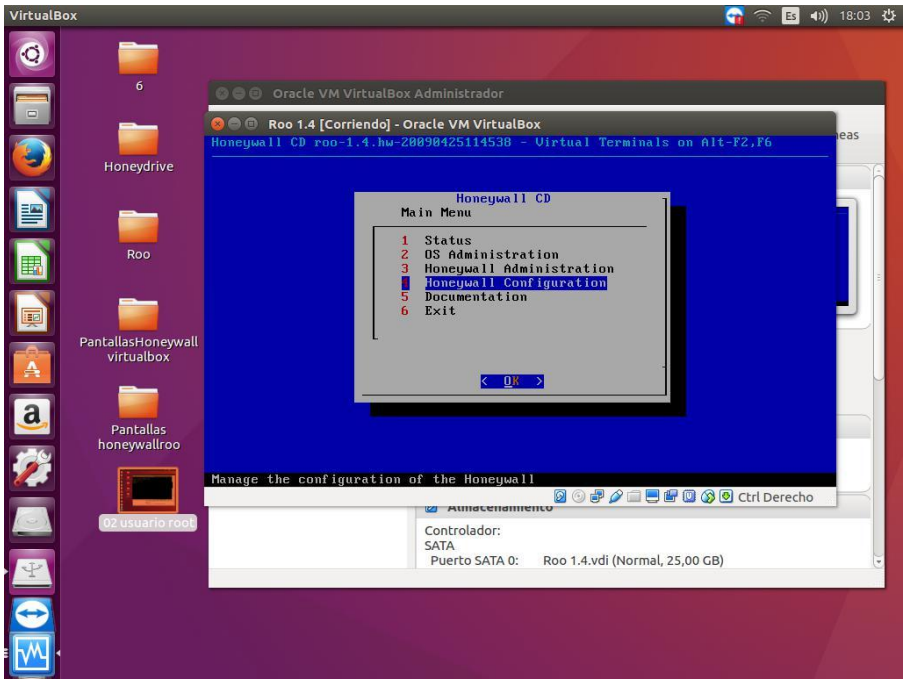

Figura 68. Menú Principal de Roo 1.4, Fuente: **Autores del Proyecto**

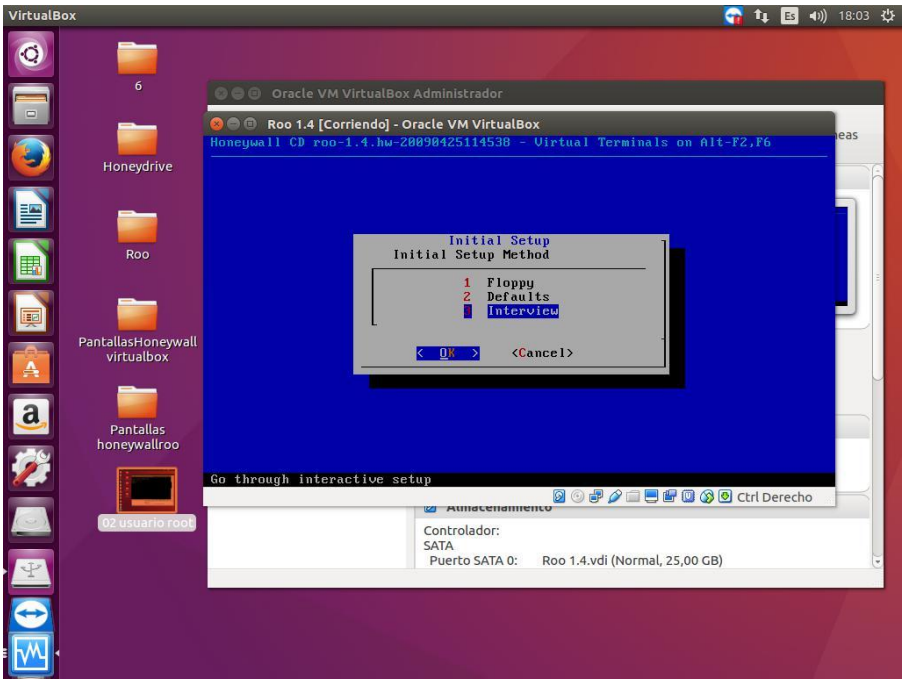

Figura 69. Menú de Modos de Configuración Inicial de Roo 1.4, Fuente: **Autores del Proyecto**

Preguntara por las direcciones IP que contienen los Honeypots, en este caso **6Guard** contiene la dirección IPv6 **1111::3/64** y a continuación la dirección de red (CIDR) donde se encuentran los Honeypots (Honeynet).

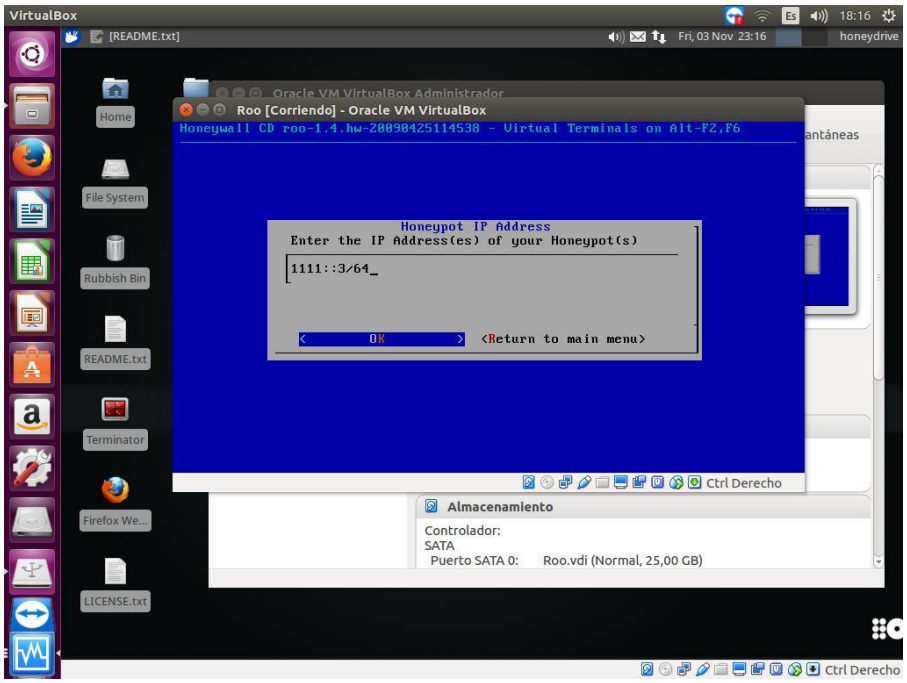

Figura 70. Dirección IP de Honeypots en Roo 1.4, Fuente: **Autores del Proyecto**

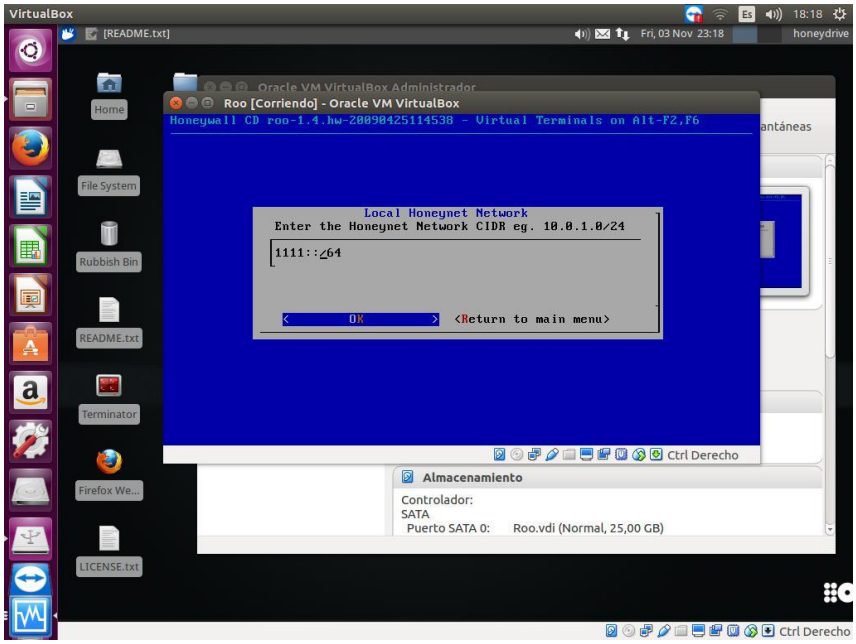

Figura 71. Dirección de Red de Honeypots en Roo 1.4, Fuente: **Autores del Proyecto**

Luego se pedirá las direcciones IP de los equipos que podrán acceder a administrar la interfaz Walleye (Server HP **3333::3/64**) mediante la interfaz **eth2 (eno1)**. Si todo se realizó correctamente, se mostraran avisos de que se encontraron las interfaces **eth0 y eth1 (eno2 y vboxnet0)** y que **eth2** está levantada.

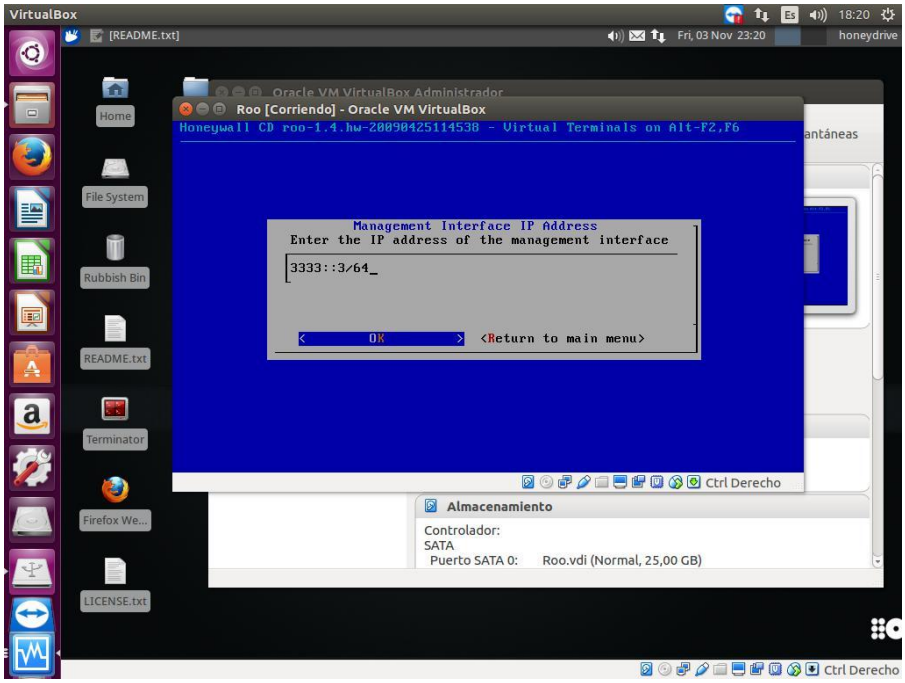

Figura 72. Direcciones IP que administraran Walleye, Fuente: **Autores del Proyecto**

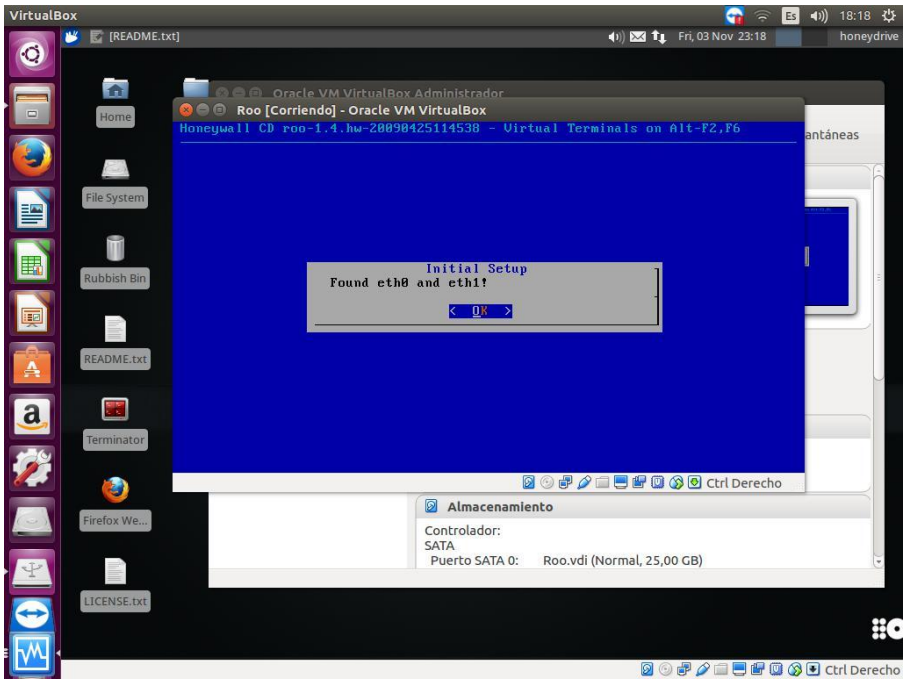

Figura 73. Interfaces eth0 y eth1 levantadas, Fuente: **Autores del Proyecto**

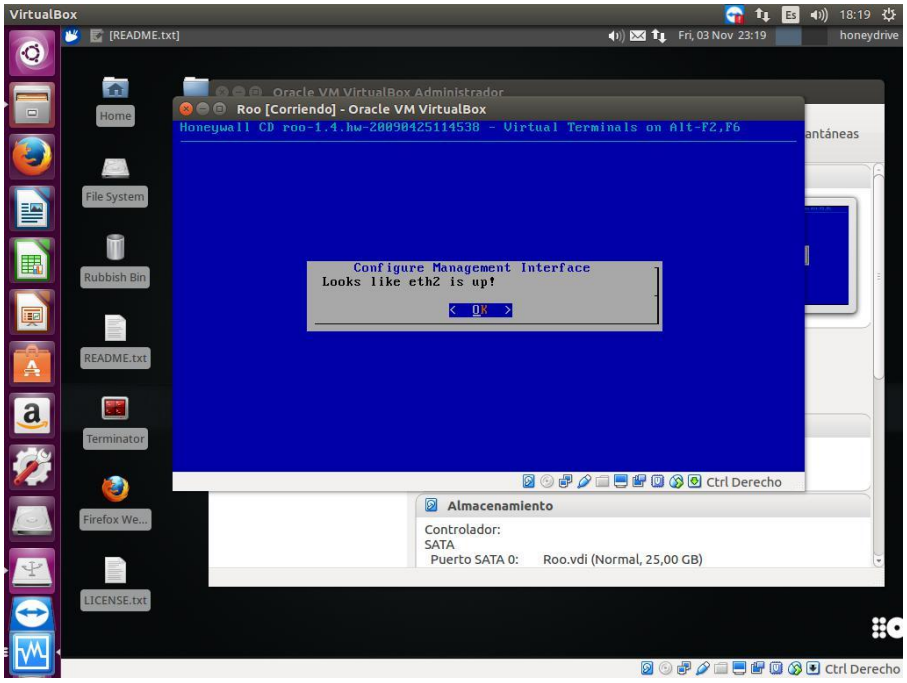

Figura 74. Interfaz eth2 levantada, Fuente: **Autores del Proyecto**

Luego se procederá realizar varias configuraciones del Honeywall Gateway tales como direcciones de servidores DNS que servirán a la red, cambios de contraseñas de los usuarios roo y root, nombres de interfaces, opciones de puertos etc… Pero es aquí donde ya hay indicios de que **Honeywall Gateway no es posible implementarse en IPv6** puesto que necesita datos para funcionar como masacraras de red de las IP antes introducidas (**en IPv6 no hay mascaras de red, pero si prefijos de red**) y direcciones Broadcast (**IPv6 no implementa Broadcast**), también sugiere el uso de Nat para los adaptadores virtuales (**IPv6 no implementa NAT).** Algunos de estos campos se dejaron en blanco o con direcciones IPv6 aleatorias.

Al terminar la configuración y volver de nuevo a cargar la terminal de comandos y entrar a los usuarios, se introdujo el comando ***ifconfig*** y se comprobó efectivamente que ninguna de las interfaces previamente configuradas tenían asignadas direcciones IP, se asume que esto sucede porque no se aceptan direcciones IP **diferentes a IPv4,** ya que si se aceptaran, las interfaces estarían configuradas correctamente y su información de direccionamiento IP seria asignada y mostrada en pantalla.

Una guía de configuración completa de Honeywall Gateway IPv4 puede verse en la presentación “Honeynet concepts and Honeywall Installation” de **Julia Yu-Chin Cheng**.

Enlace: https://es.slideshare.net/YuChinCheng/honeywall-roo-1 (Yu-Chin Cheng, 2015)

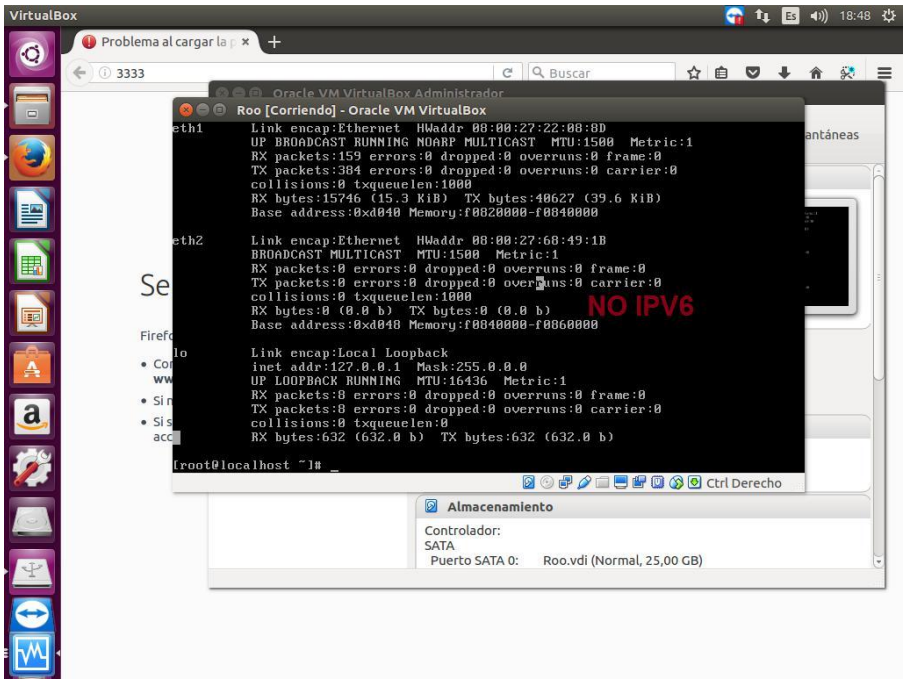

Figura 75. Interfaces sin direcciones IPv6 asignadas, Fuente: **Autores del Proyecto**

Por consiguiente queda expuesta la incompatibilidad entre el CDROM Honeywall Roo 1.4 y el protocolo IPv6 o que aún no se evidencia documentación de su implementación. Dado lo anterior es importante mencionar que por el momento no es posible la implementación de Honeywall ante el protocolo de Internet versión 6 (IPv6). Sin embargo SNORT, su pieza central si se cuenta con documentación oficial para poder implementarse bajo el protocolo IPv6 e incluso tiene un proyecto reciente para la implementación de un Plugin con soporte añadido y mejorado para IPv6.

SNORT

SNORT es un **Sistema de Detección de Intrusos (IDS)** basado en red **(IDSN)** desarrollado por **CISCO Systems** y de Código Abierto. Dispone de un lenguaje de creación de reglas en el que se pueden definir los patrones que se utilizarán a hora de monitorizar el sistema. Además, ofrece una serie de reglas y filtros ya predefinidos que se pueden ajustar durante su instalación y configuración.

Se trata de un sistema basado en red que monitoriza todo un dominio de colisión y funciona detectando usos indebidos. Estos usos indebidos (o sospechosos) se reflejan en una base de datos formada por patrones de ataques. Dicha base de datos se puede descargar también desde la propia página web de SNORT, donde además se pueden generar bases de patrones "a medida" de diferentes entornos (por ejemplo, ataques contra servidores web, intentos denegaciones de servicio, Exploits...). (LinuxParty, 2010)

Como se dijo anteriormente, SNORT **si tiene soporte para IPv6**, pero tal soporte y su cantidad de reglas en este protocolo es bastante débil con respecto a su soporte para el protocolo IPv4, sin embargo para este proyecto es más que suficiente con sus herramientas y su compatibilidad con IPv6. Cabe notar que existe un proyecto llamado **The IPv6 SNORT Plugin,** el cual agrega reglas de detección y un preprocesador para el Protocolo de descubrimiento de vecinos. Está dirigido a la detección de actividad sospechosa en redes IPv6 locales y puede detectar elementos de red mal configurados, así como actividades maliciosas de los atacantes en la red**, pero tal Plugin no se implementara en este proyecto dado a su complejidad, su poca información y resultados sin verificar.**

Su documento se puede recuperar del siguiente enlace:

https://www.ernw.de/download/20140318_Troopers14_Snort_IPv6.pdf

Tampoco es necesario para este proyecto crear una base de datos MySQL ya que la información de Logs de SNORT será leída por un Sniffer de red como Wireshark. SNORT cuenta también con dos modos de operación opcionales: como Sniffer y como Logger de Paquetes, puesto que su modo IDSN (IDS) ya lleva estas opciones incluidas y posibilita la lectura de Sniffers externos, se configurara bajo modo IDSN el cual es el más completo.

Figura 76. Logo de SNORT, Fuente: **snort.org**

Preparación y Topología

Es necesario saber que SNORT distingue dos redes primarias, una es la **Red de Hogar (Home Net)** la cual es la que contendrá todas las direcciones de red de los equipos que se van a proteger (alertas y detecciones), en este caso los rangos IPv6: **1111::/64 y 2222::/64**, y la **Red Externa (External Net),** la cual es la que contiene **todos los rangos de red de equipos que no pertenecen a la Red de Hogar** y por lo tanto los que se consideran como intrusos, para este caso **3333::/64**.

También es de gran utilidad utilizar la función **Puerto Espejo (Port Mirroring),** de los Switch CISCO, cuya finalidad es realizar una copia del tráfico recibido por un puerto y enviarla directamente al puerto por donde escucha SNORT. La función Puerto Espejo puede enviar tanto

información de un solo equipo directamente conectado, como de una VLAN (para varios Equipos).

Así que se utilizara la siguiente **Topología LAN Honeynet**, donde el **PC Intruso** se encuentra en la misma red donde esta SNORT y 6Guard; segmentada por el **Switch 1**. En el caso de estar el Intruso del **otro lado de la red** (Switch 2, donde esta DionaeaFR-IPv6), es obligatorio crear una VLAN, conectar ambos Switch (modo Trunk) y hacer Puerto Espejo de este para enviar al SNORT. **Como el servidor Hewlett Packard posee dos interfaces**, se utilizara la **eno1 para la conexión de SNORT con el resto de la red**, (y la eno2 para el Honeypot 6Guard) La Versión Utilizada de SNORT es la 2.9.7 la cual ya viene con **IPv6 activado**, el sistema operativo utilizado es Ubuntu 16.04.

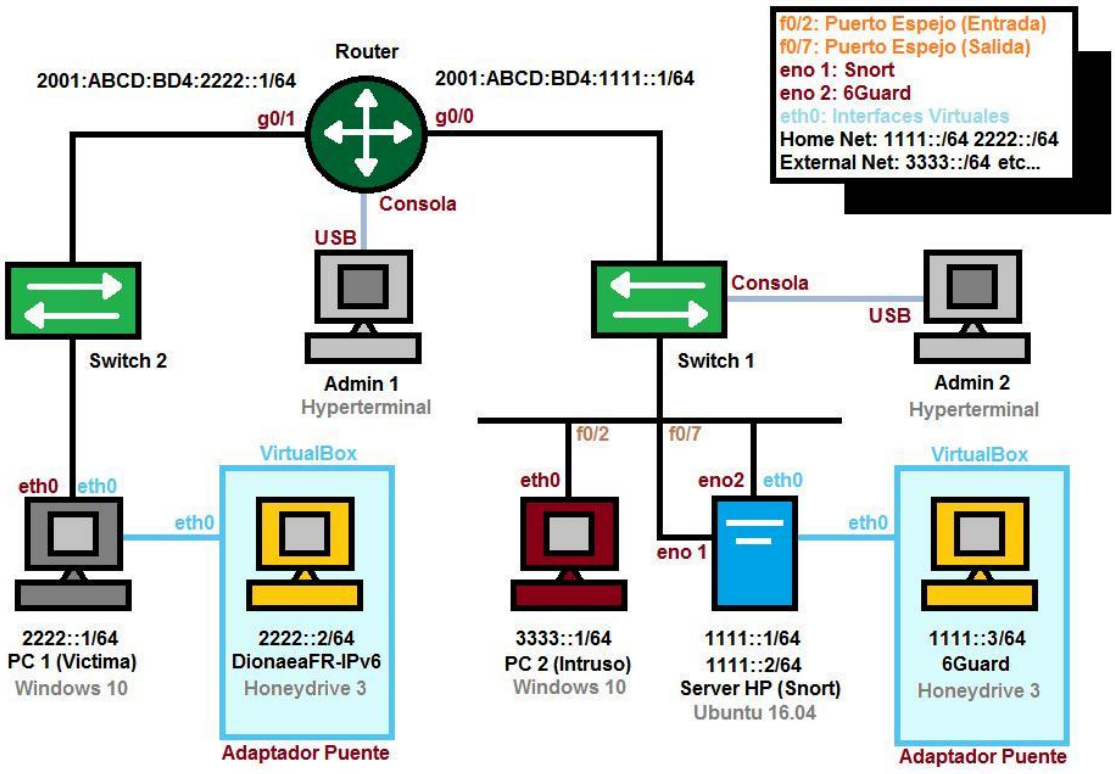

Figura 77. Topología LAN Honeynet IPv6 de Prueba SNORT, Fuente: **Autores del Proyecto**

Configuración del Switch 1, Puerto Espejo

La Función Puerto Espejo puede ser implementada en los Switch utilizados en el Laboratorio de Redes y Telecomunicaciones de la UFPSO, estos son de la marca **CISCO serie 2960 Plus**. Por medio de un PC se administrara el Switch 1 (puerto Consola) y se implementara esta función. La siguiente imagen es una topología estándar del funcionamiento de la función Puerto Espejo para ser empleada a SNORT:

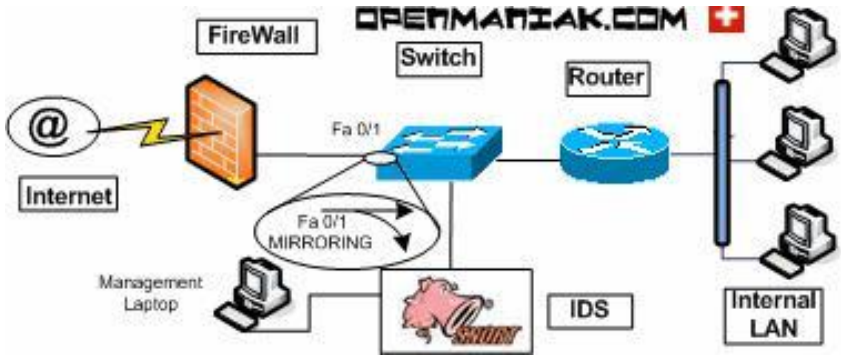

Figura 78. Topología de Puerto Espejo con SNORT, Fuente: **openmaniak.com**

Las interfaces con las que cuenta el Servidor HP, SNORT escuchara directamente por la eno1 al realizarse la función de Puerto Espejo en el Switch 1.

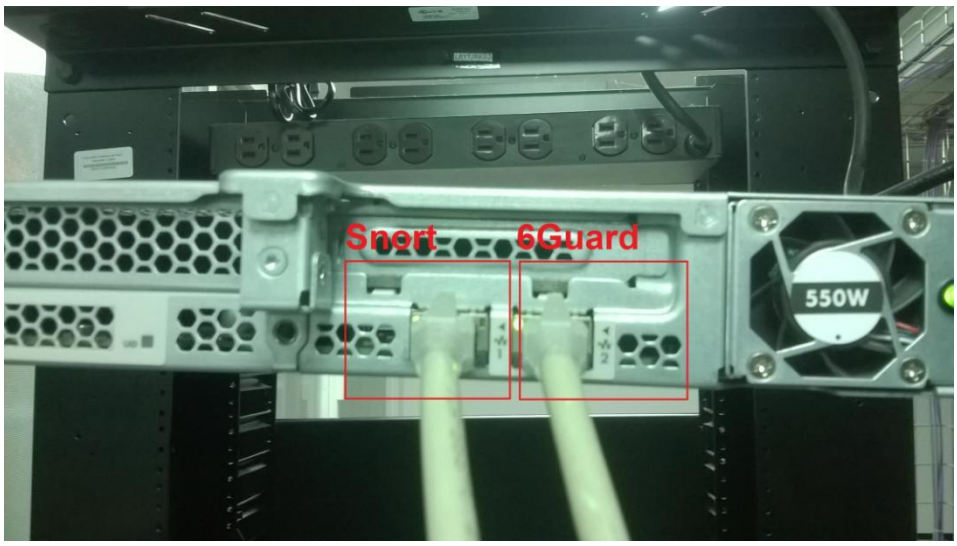

Figura 79. Interfaces eno1 y eno2 del Server HP, Fuente: **Autores del Proyecto**

Las interfaces del Switch 1 que interactúan para la función de Puerto Espejo son **f0/2** a la cual se le realizará una copia de su tráfico y se enviara por **f0/7** que va conectada a la interfaz eno1 del Servidor HP por donde SNORT escuchara directamente. (**f0/1** a g0/0 Router y **f0/5** a eno2 6Guard)

Figura 80. Interfaces f0/2 y f0/7 del Switch 1, Fuente: **Autores del Proyecto**

Una Vez dentro del Switch 1 se entrara al modo de configuración global y mediante el comando ***monitor sesión*** se configurara **f0/2 como interfaz de entrada (fuente) y f0/7 como interfaz de salida (destino)**, y posteriormente se levantaran con el comando ***no shutdown (no sh)*** para asegurase de que estén totalmente operacionales.

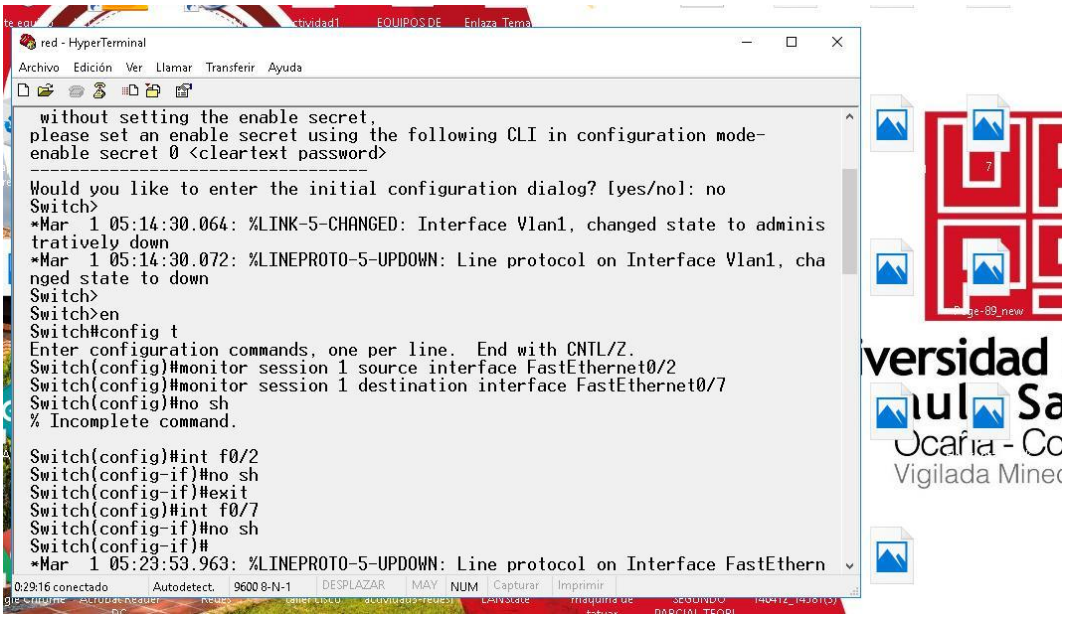

Figura 81. Configuración de Puerto Espejo en el Switch 1, Fuente: **Autores del Proyecto**

Configuración del Router, Enrutamiento Estático IPv6

Actualmente SNORT también es capaz de proteger equipos remotos en los cuales no está instalado. En nuestra **topología LAN Honeynet** segmentada por dos Switch, se instalara SNORT en el Servidor HP que cuenta con la dirección 1111::1/64 (eno1), **se protegerá la red 1111::/64** y también **se protegerá la red 2222::/64** la cual contiene el PC victima (2222::1/64) y el Honeypot DionaeaFR-IPv6 (2222::2/64) y que además se encuentra del otro lado de la red segmentada por el **Switch 2 (g0/1 del Router).**

Para esto es necesario configurar **Enrutamiento IPv6 en el Router**, debido a que la red no es extensa, se optó por utilizar **Enrutamiento Estático IPv6.** Con todo previamente configurado (direcciones IPv6, puertos, *IPv6 unicast-routing*), este modo de Enrutamiento se configura con el comando ***IPv6 route [dirección destino]/prefijo [dirección fuente o puerto de conexión conocido]***. Para conectar una dirección desconocida al puerto g0/0 del Router como lo es la 2222::/64 se introduce la siguiente línea de código en la configuración global del Router: ***IPv6***

route 2001:ABCD:BD4:1111::0/64 2222::0 , De esta forma la red segmentada por el Switch 1 tiene comunicación a la red segmentada por el Switch 2 y viceversa, pudiendo así SNORT **proteger cualquier equipo que se desee de toda la red Honeynet.**

Con el comando ***show IPv6 route static*** se mostrara la información y las conexiones del Enrutamiento Estático IPv6 utilizado. La Letra **S** mostrara que interfaces están conectadas por medio de Enrutamiento Estático a un rango de dirección de red específico.

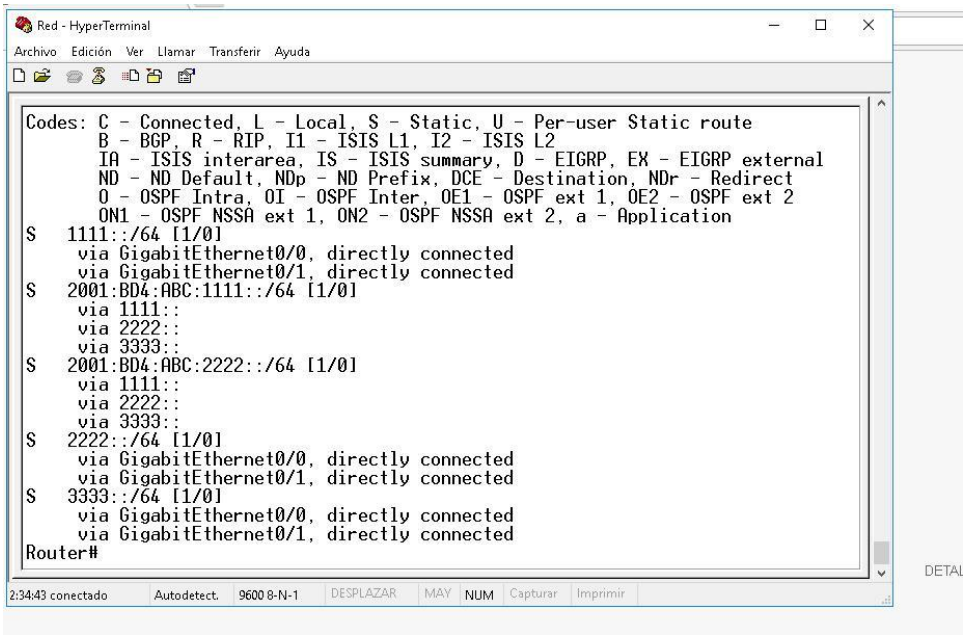

Figura 82. Router**,** Enrutamiento Estático IPv6, Fuente: **Autores del Proyecto**

Mediante el uso de comandos **ping** se puede confirmar la conexión entre los dispositivos de la red Honeynet. Es necesario **desactivar el Firewall** de las maquinas o **crear reglas de entrada dentro de este** que permitan tráfico IPv6 (TCP, UDP, **ICMPv6** etc…). Como este proyecto está enfocado a la recolección de la información del atacante, **se recomienda desactivar todo tipo de Firewall**, mientras que en otros proyectos sería más aconsejable la creación de reglas.

Los comandos para emplear ping en el protocolo IPv6 (ICMPv6) en los distintos sistemas operativos son los siguientes:

- ***ping6*** sirve para realizar ping a direcciones IPv6 desde Linux. (Ubuntu, Honeydrive, Kali)

- ***ping -6*** sirve para realizar ping a direcciones IPv6 desde Windows.

- ***ping IPv6*** sirve para realizar ping a direcciones IPv6 desde Hyperterminal. (Router)

Nota: Es posible que se tome como dirección de red preferida la dirección de enlace local en vez de la dirección global, en el caso de no hacer ping a una dirección global pero si a una de enlace local sigue existiendo conexión, sobre todo si participa una máquina virtual.

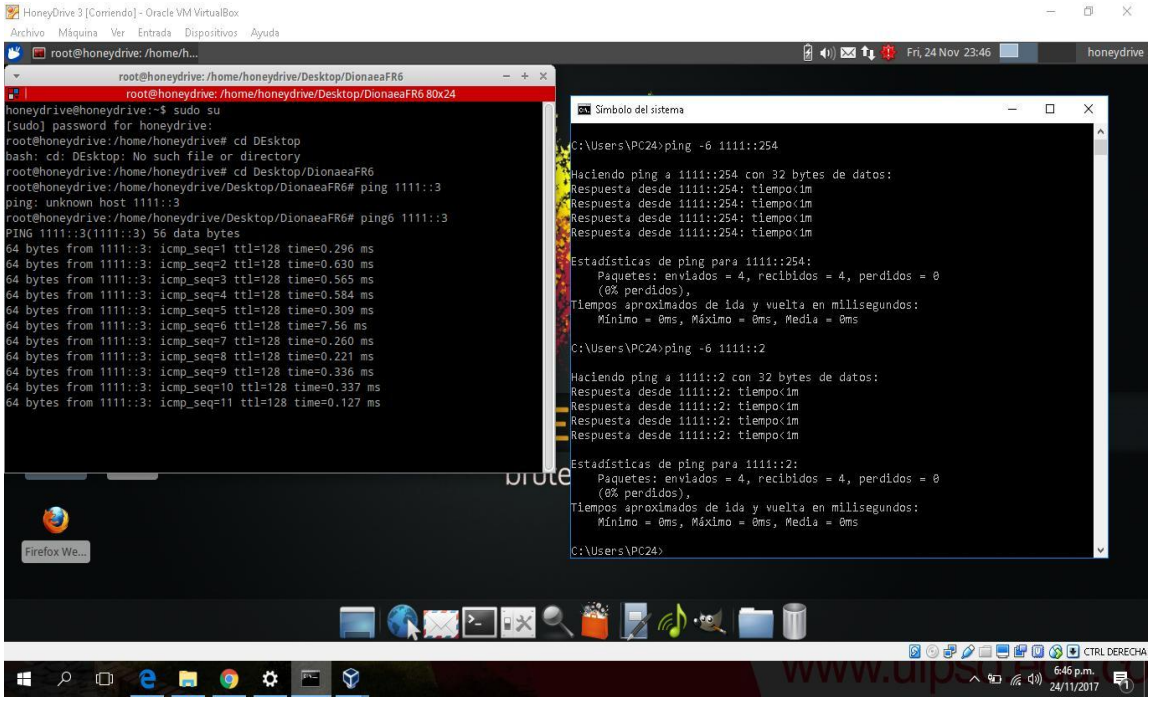

Figura 83. Pruebas de Ping IPv6 exitosas (Linux y Windows), Fuente: **Autores del Proyecto**

Instalación y Configuración Primaria de SNORT

Para instalar SNORT en Ubuntu se entrara la Terminal, pero antes se necesitara descargar todas las librerías necesarias para sus correcto funcionamiento, debido a que la versión de Ubuntu utilizada es la 16.04 ya cuenta con varias de estas librerías instaladas y actualizadas, sin embargo como se especifica en su manual de configuración el comando con las librerías necesarias el siguiente:

sudo apt-get install -y gcc libpcre3-dev zlib1g-dev libpcap-dev openssl libssl-dev libevent-dev libdumbnet-dev libedit-dev bison flex libdnet libreadline-dev libtool automake

Lugo se procederá a descargar la última versión disponible de SNORT mediante el comando ***apt-get install snort.*** Una vez se termine se introducirá el comando ***dpkg-reconfigure snort*** para realizar la configuración primaria adecuada de SNORT. La primera pantalla mostrara la interfaz de escucha por donde SNORT recibiría el tráfico proveniente, en este caso **eno1**:

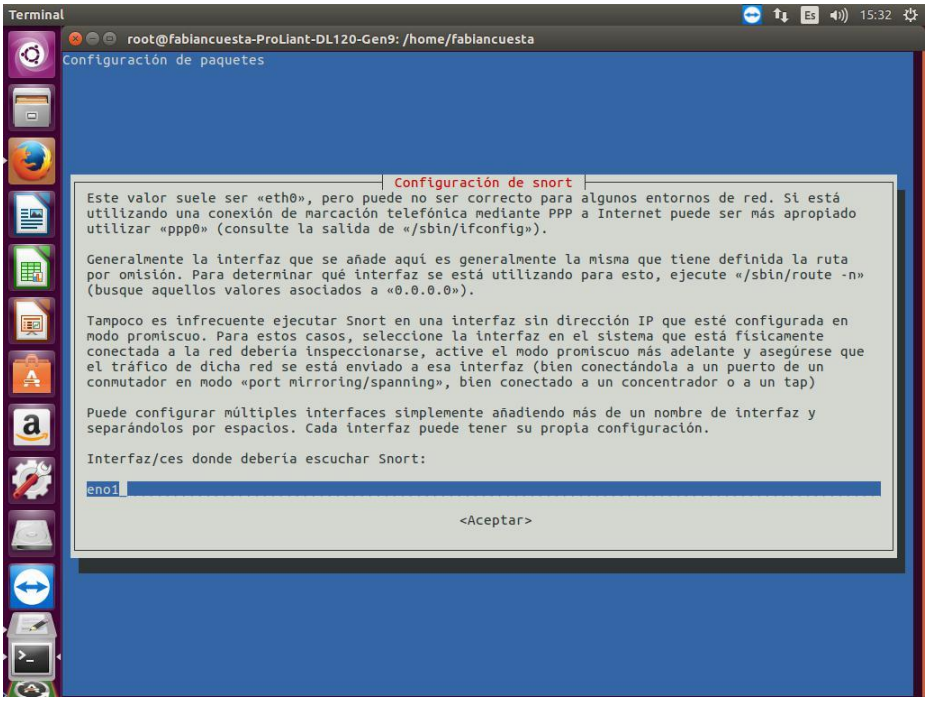

Figura 84. Configuración de la Interfaz de escucha de SNORT, Fuente: **Autores del Proyecto**

A continuación se introducirán las direcciones de red a proteger, las cuales serán incluidas dentro de la Red de Hogar (Home Net), en este caso **1111::/64 (Local) y 2222::/64 (Remota)**

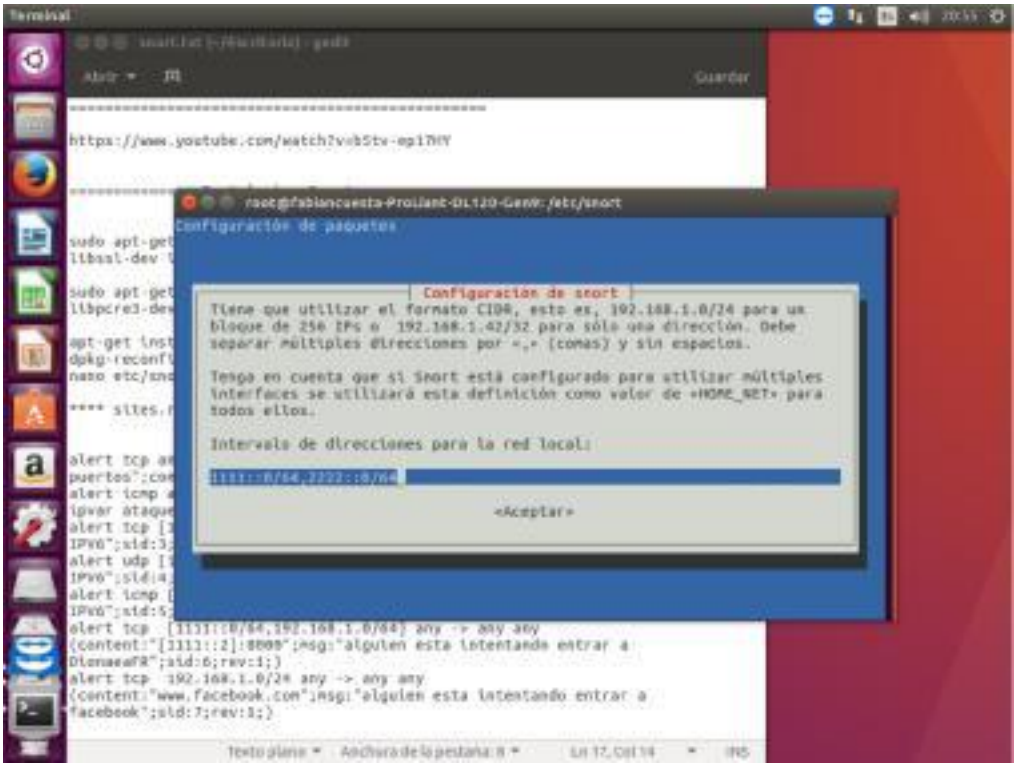

Figura 85. Redes IPv6 Protegidas en la Red de Hogar de SNORT, Fuente: **Autores del Proyecto**

Y se tendrá que configurar todas las variables básicas de SNORT, tales como el modo de arranque, el cual se realizara de forma manual, el modo promiscuo habilitado (para que reciba información de toda la red), informe de alertas vía email (en caso de ser red conectada a internet) etc…

Al finalizar esta configuración básica, el servicio de SNORT se reanudara y luego se procederá a realizar la configuración avanzada y especifica con los parámetros y reglas de IPv6 necesarios para que funcione adecuadamente SNORT.

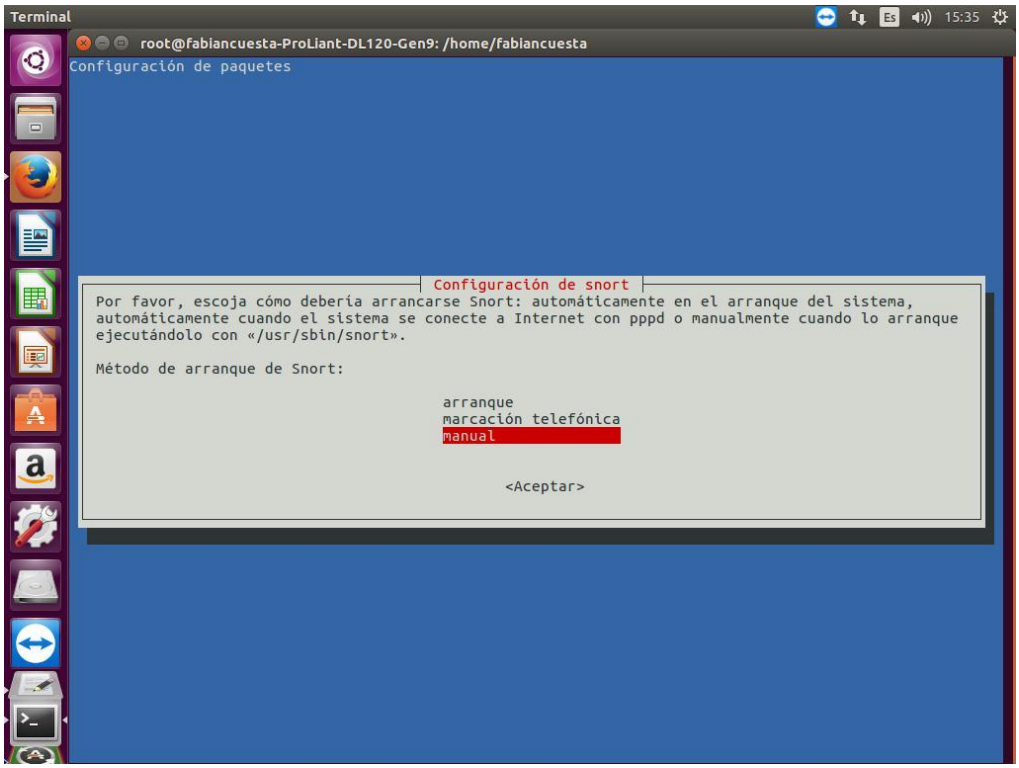

Figura 86. Configuración para Arranque Manual de SNORT, Fuente: **Autores del Proyecto**

Configuración Avanzada IPv6 y Creación de Reglas IPv6

A continuación se procederá a editar el archivo **snort.conf** el cual contiene los parámetros de configuración avanzada y de conexión. SNORT se instala en la carpeta **etc** (***cd etc/snort***) y con el comando ***nano*** se accede y se edita el archivo mencionado (***nano snort.conf***)

En **el paso 1** se establecerán las variables de red, aquí se confirma que la línea ***ipvar HOME_NET*** contenga las direcciones a proteger y definir que la línea ***ipvar EXTERNAL_NET*** sea todo lo que no es la HOME NET (***ipvar EXTERNAL_NET !$HOME_NET***).

También es necesario introducir los comandos que iniciaran las alertas en el protocolo IPv6 mediante el siguiente parámetro ***IPv6_frag <option1 arg1>[, <option2 arg2>, ...],*** esto es:

bsd_icmp_frag_alert on Iniciara las alertas del protocolo ICMPv6 fragmentadas en el procesador (BSD) que serán vulnerables.

bad_IPv6_frag_alert on alerta si el segundo paquete ha sido visto por sí mismo.

Enlace de Instrucciones de SNORT: https://www.snort.org/faq/readme-IPv6

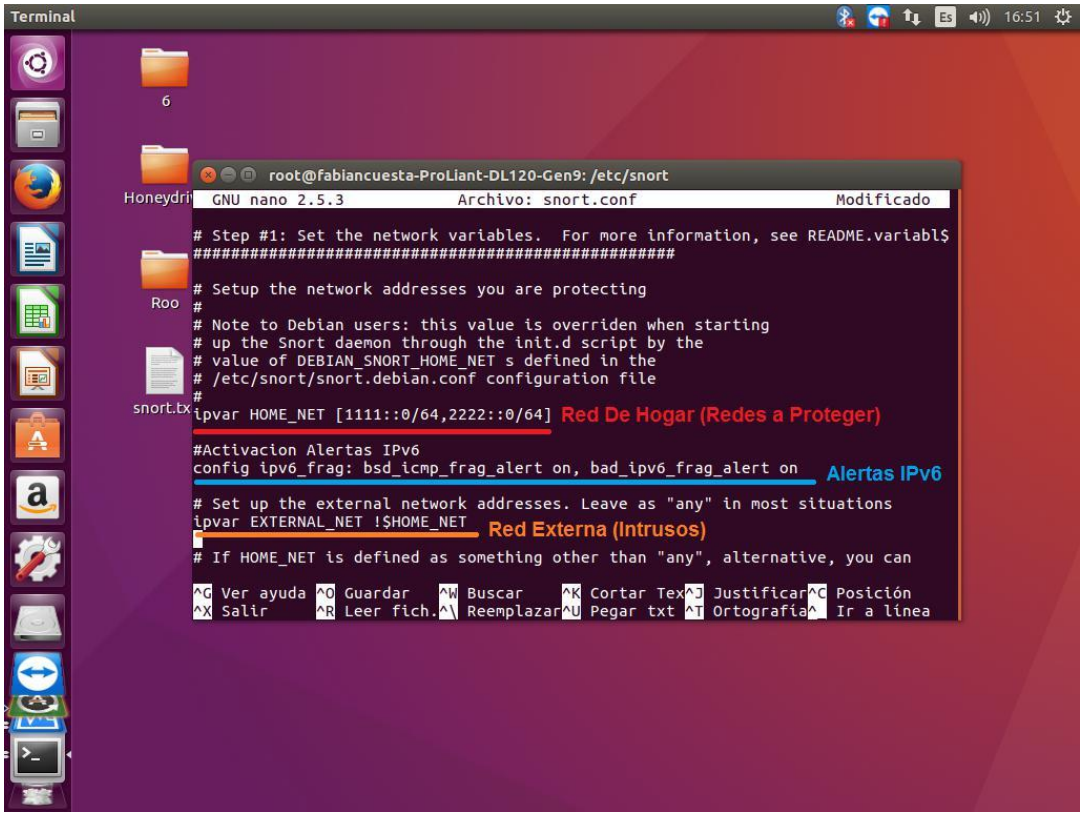

Figura 87. Variables de Red IPv6 en el archivo snort.conf, Fuente: **Autores del Proyecto**

Las otras configuraciones de los otros pasos se dejaran así, pero en **el paso 7** es donde se hace la inclusión de reglas que SNORT tiene por defecto en su librería de reglas ubicada en la carpeta **etc/snort/rules**. Todas las reglas que están allí son para el protocolo IPv4 y la mayoría para entornos de Internet, así que es necesario incluir en este paso una línea de código que invoque las reglas IPv6 que se añadirán después en un archivo llamado **sites.rules,** el cual se creara después.

Se escribirá entonces la línea ***include $RULE_PATH/sites.rules*** y se guardaran (ctrl+X) los cambios realizados al archivo snort.conf.

Cabe destacar que todas las configuraciones que se hagan, edición y creación de archivos y/o carpetas, se harán desde la terminal por medio del **súper usuario root (sudo su)**, ya que el **directorio etc está protegido** y no se puede editar como usuario normal.

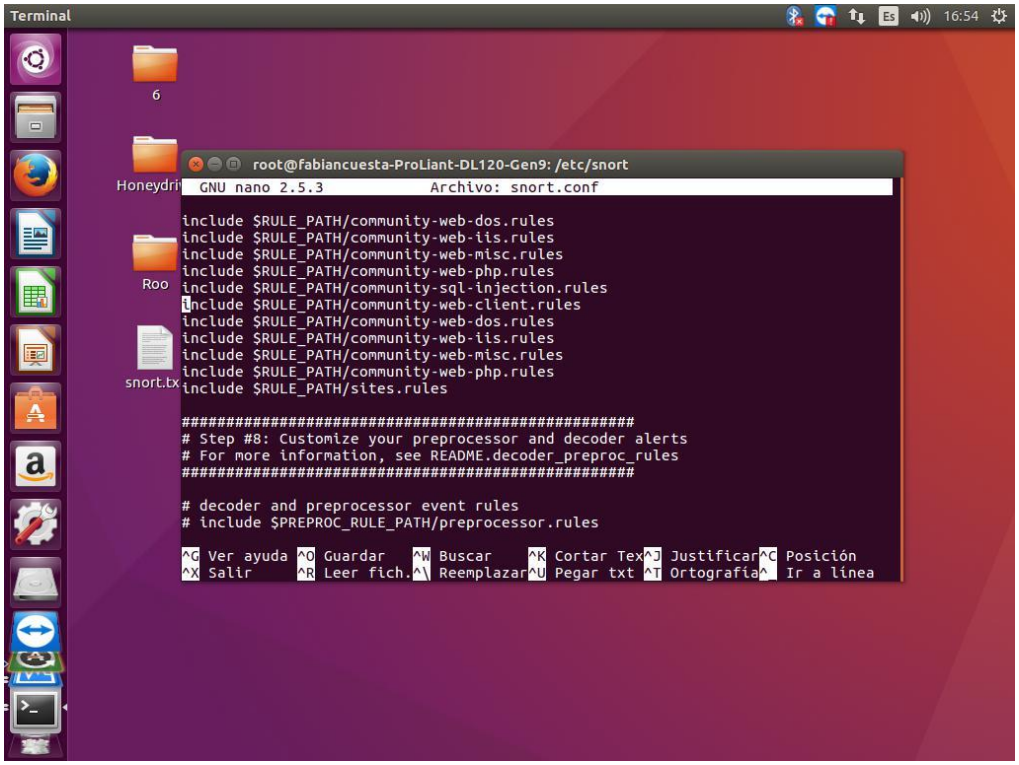

Figura 88. Invocación del archivo sites.rules en el archivo snort.conf, Fuente: **Autores del Proyecto**

Se ingresara a la carpeta **rules** (***cd rules),*** y se creara el archivo **sites.rules (*nano sites.rules*)**, a continuación se escribían las reglas con los siguientes parámetros

alert [protocolo] [red de entrada] [puertos] -> [red afectada] [Puertos] (msg:"mensaje";sid:100000+;rev:1;)

De esta forma si se quiere alertar todo intento de acceso TCP de la Red Externa a cualquier puerto de la Red de Hogar:

alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"Intento de Acceso No Autorizado IPv6";sid:100001;rev:1;)

También se pueden escribir direcciones de red por individual y crear variables con parámetros para invocarlas luego, esto si se quiere ser más específico de diferenciar las alertas, una que alerte un acceso o ping desde determinada red o desde la Red de Hogar y otra alerta lo que venga de la Red Externa, este tipo de regla se incluyó también en el archivo sites.conf pero se dejó comentarizada por medio de la **etiqueta inicial #.**

Se dejó funcional la detección por medio de alertas de la red Externa de los protocolos TCP e ICMP (ICMPv6 en este caso). El archivo sites.rules ya fue incluido antes de su creación en el archivo de configuración de SNORT (snort.conf).

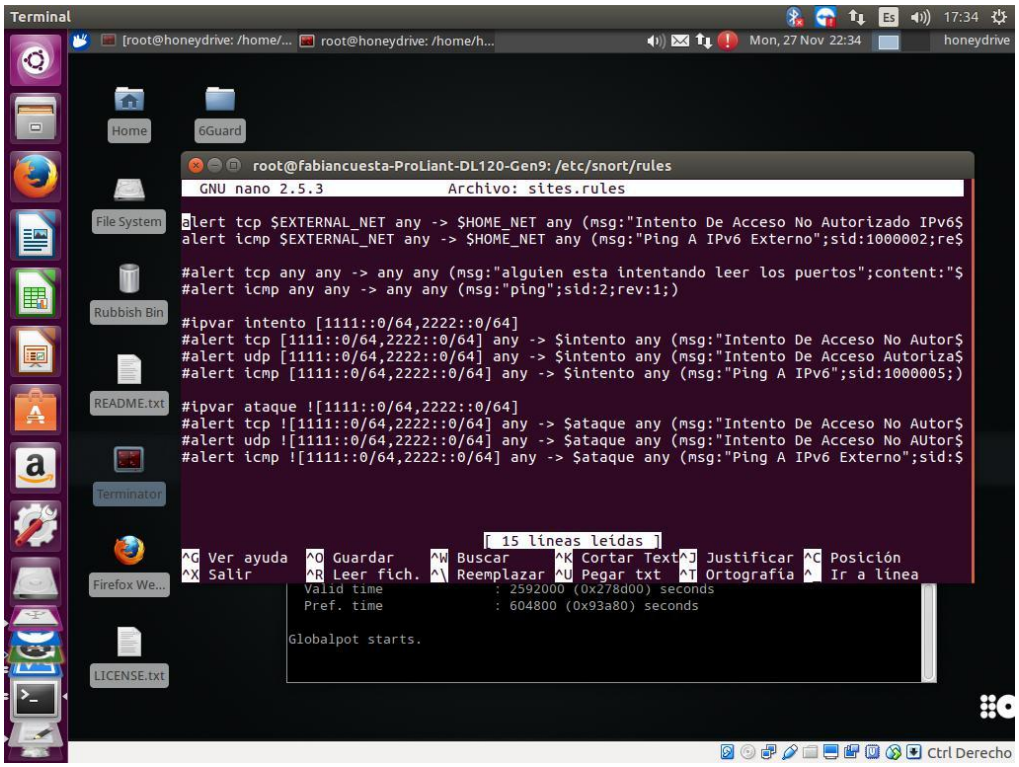

Figura 89. Reglas propias para IPv6 en el archivo sites.rules, Fuente: **Autores del Proyecto**

Cabe destacar que el proyecto **The IPv6 SNORT Plugin,** tiene sus propias reglas IPv6 las cuales son más eficaces, más específicas pero aún no han sido implementadas a la versión más reciente de SNORT, por lo que si se intentan incluir en este archivo, no funcionaran, terminara con error al momento de ejecutarse.

Ejecución de SNORT

Para ejecutar SNORT en su modo IDSN (el más complejo y completo) es necesario dar la siguiente orden con el siguiente parámetro (estando en la carpeta etc/snort):

snort -A [modo de registro] -c [archivo de configuración] -i [Interfaz de escucha]

Por lo que se ejecutara de la siguiente forma:

snort -A console -c snort.conf -i eno1

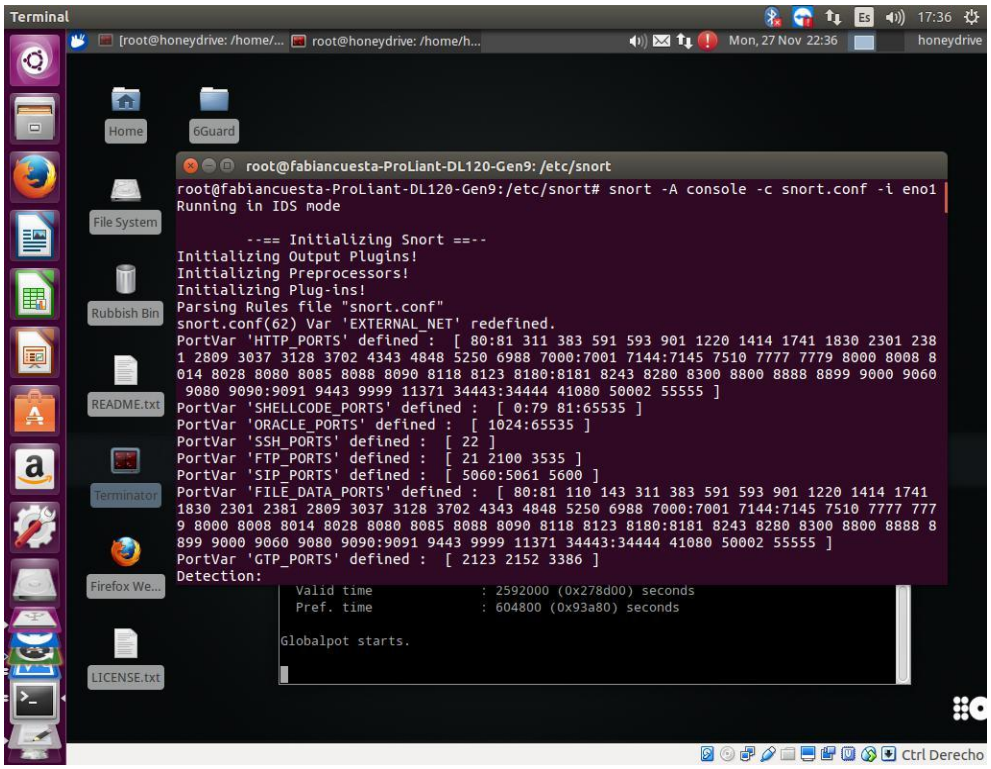

Figura 90. Ejecución de SNORT modo IDSN (IDS), Fuente: **Autores del Proyecto**

Se realizaron **dos pruebas de ejecución previas a la ejecución total de SNORT**, **la primera sin Puerto Espejo** y **la segunda desde Honeydrive** con su versión más estable 2.9.2 **sin enrutamiento**, con el **fin de darle prioridad completa a IPv6 en un ambiente donde existen también redes IPv4**.

Ejecución de SNORT sin Puerto Espejo

Para esta prueba debido a que no se realizó la función de Puerto Espejo, SNORT detectaba todo tipo de tráfico proveniente del Router, esto es que las detecciones intrusos las reconocía como si fuera el mismo Router, ya que todo el tráfico de la red Honeynet tiene que primero pasar obligatoriamente por el Router. SNORT siempre está escuchando alertas de tráfico de envíos de paquetes a puertos y pings continuos del Router a la red. Se realizaron intentos de acceso TCP (al puerto 8000) al servidor Ubuntu 1111::1/64 (Donde se encuentra SNORT Instalado) desde el equipo intruso 3333::1/64; como también pings de IPV6. Para ambos casos en medio de las alertas de tráfico continuo, se registraron estas alertas pero como si fueran hechas por el Router. Las siguientes imágenes muestran primero como se registró acceso TCP de 3333::1 al puerto 8000 del Servidor HP (1111::1) y luego un Ping IPv6 (ICMPv6) del PC Intruso (3333::1) al servidor HP.

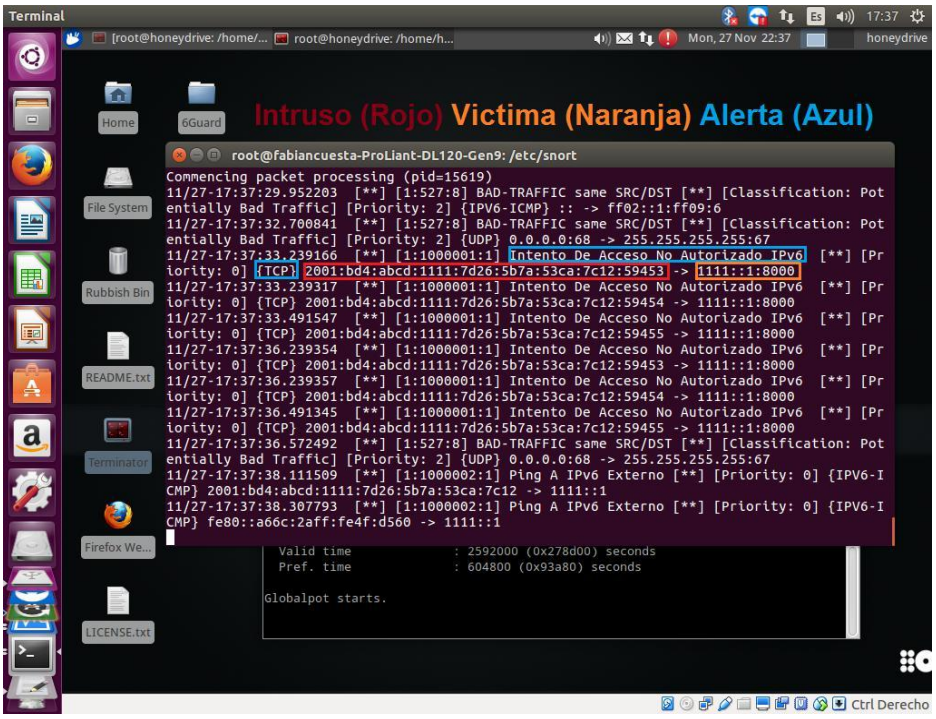

Figura 91. Detección de Alerta TCP local, Sin Modo Espejo, Fuente: **Autores del Proyecto**

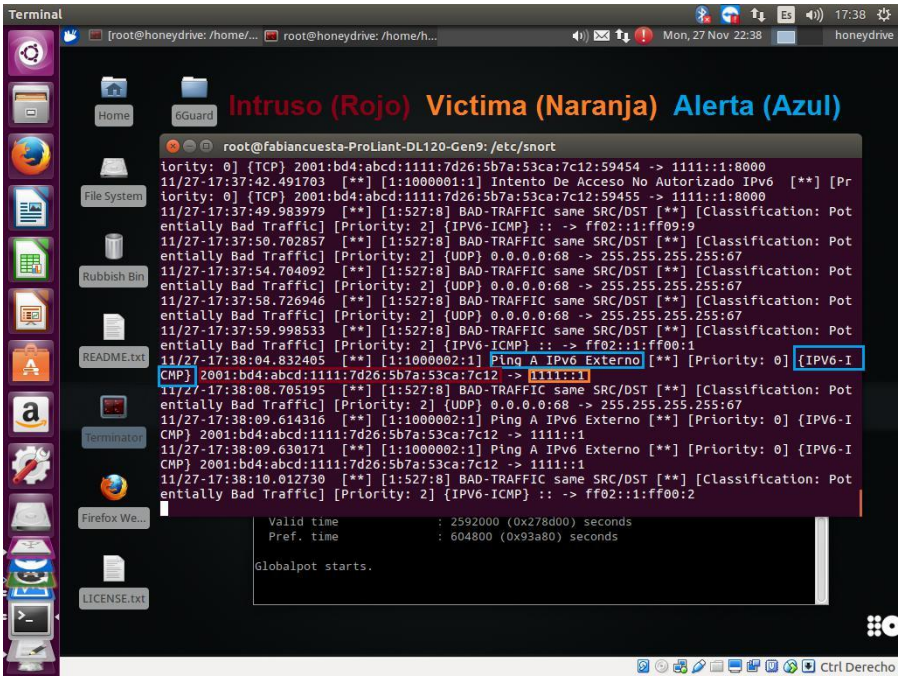

Figura 92. Detección de Alerta ICMPv6 (Ping) local, Sin Modo Espejo, Fuente: **Autores del Proyecto**

Ejecución de SNORT 2.9.2 en Honeydrive (Con Prioridad de IPv6 sobre IPv4)

Como este proyecto está todo realizado en IPv6, no sería posible saber con qué prioridad actúa IPv6 con respecto a IPv4 en SNORT, sin embargo es posible anular la detección de redes IPv4 parcial o totalmente y darle exclusividad a las redes IPv6. Para esto se probó en una red donde existía un PC1 Windows con dirección IPv4 y acceso a internet (192.168.1.6/24) y que por medio de una regla manual, solamente se detectaría cuando este entrase a Facebook, y un PC2 Windows con máquina virtual corriendo Honeydrive donde solo se protegió la dirección de la máquina virtual (1111::2/64) dejando al PC2 (1111::3/64) como intruso. Tampoco se realizó Enrutamiento.

Honeydrive posee ya todas las librerías necesarias para la rápida instalación y ejecución de SNORT, con la posibilidad también de instalarlo desde cero (***make install***). En Honeydrive se descargó la versión 2.9.2, esta no venía con IPv6 activado pero se podía activar mediante la línea ***--enable-IPv6*** durante la instalación desde cero (*make install*) de SNORT o con la inclusión de tal línea en el archivo de configuración; snort.conf.

De esta forma se le dio Prioridad completa a IPv6 analizando exclusivamente todo el tráfico de este protocolo y la configuración del preprocesador en modo IPv6. Como se mencionó anteriormente, manualmente se incluyó una regla para que solamente se activara una alerta IPv4 cuando el PC1 entrara a Facebook como lo muestran las siguientes imágenes:

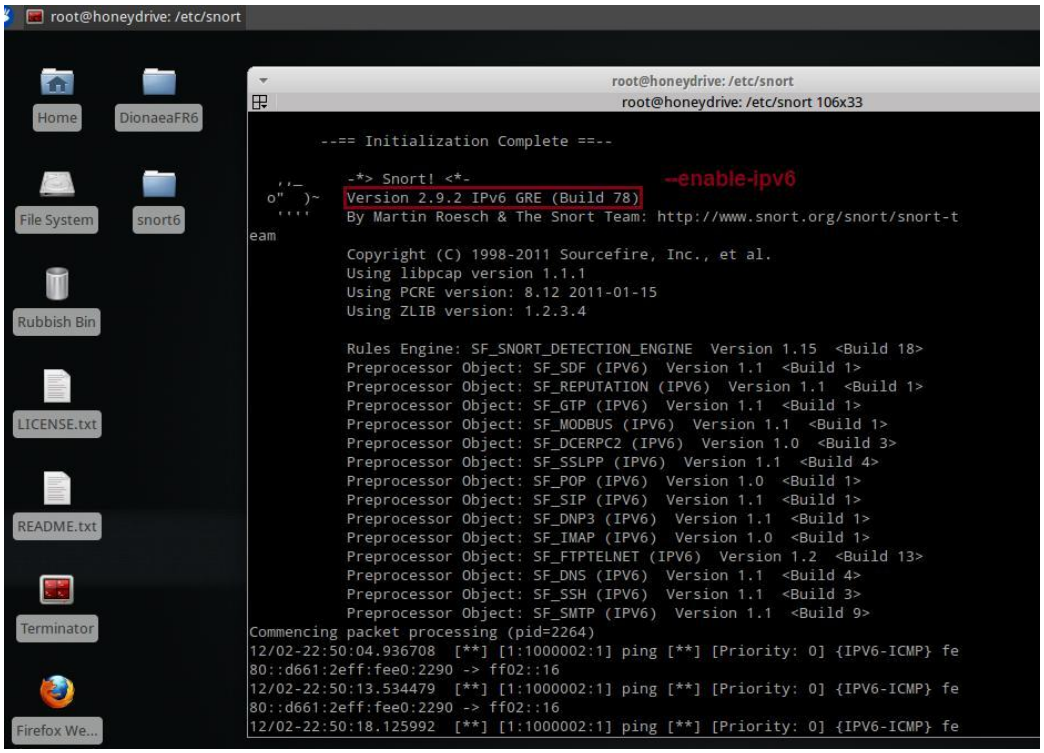

Figura 93. Configuración Total IPv6 en SNORT 2.9.2, Fuente: **Autores del Proyecto**

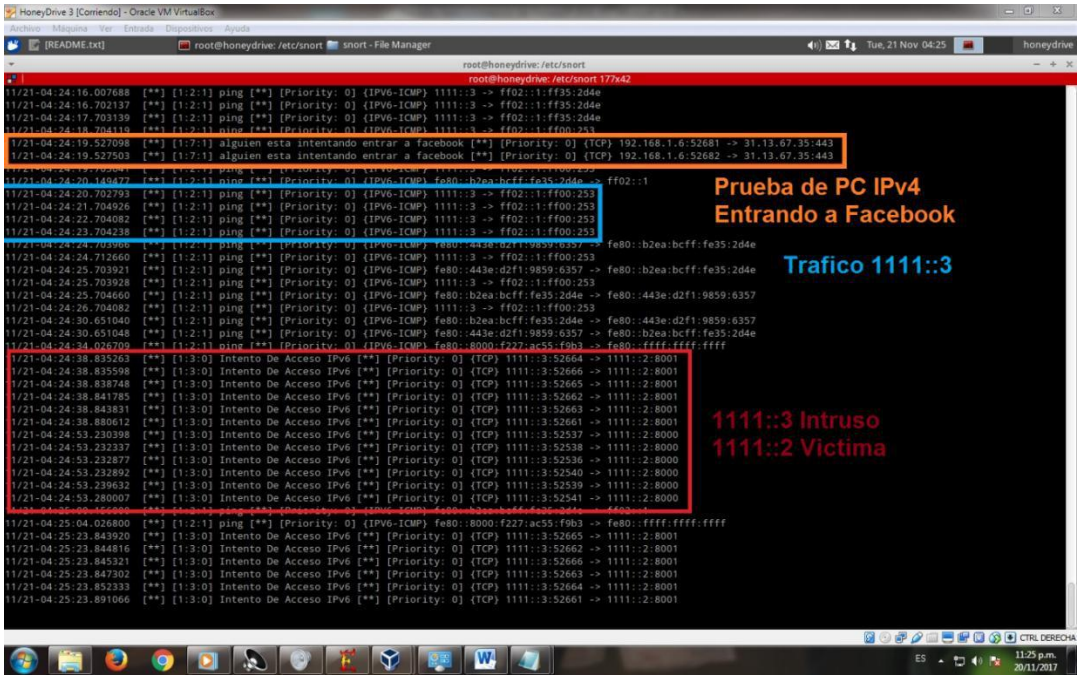

Figura 94. IPv6, Alertas y Prioridad sobre IPv4 en SNORT 2.9.2, Fuente: **Autores del Proyecto**

Ejecución Total de SNORT en la red LAN Honeynet

Una vez realizada las dos pruebas anteriores, se procederá a realizar la ejecución total de SNORT 2.9.7 (su última versión) con la topología de red LAN Honeynet diseñada anteriormente y con todos los parámetros establecidos, Enrutamiento Estático IPv6 y Puerto Espejo. Además esta vez se intentara acceder desde el PC Intruso al PC Victima ubicado del otro lado de la red y protegido remotamente por SNORT.

Para esto se le añadió automáticamente una dirección IPv6 de anuncio del Router (RA) para el PC Intruso con dirección 3333::1/64, **la cual añade a su dirección de link-local automática** (fe80:4833:462d:378e:f3d8**), la dirección global del Router por donde está conectado directamente** que es la interfaz g0/0 (2001:ABCD:BD4:1111::1/64), por lo tanto, como lo muestra la siguiente imagen, se sabe que la dirección **2001:ABCD:BD4:1111:4833:462d:378e:f3d8 hace referencia al PC Intruso** con dirección IPv6 global 3333::1/64 en su tráfico hacia el Router.

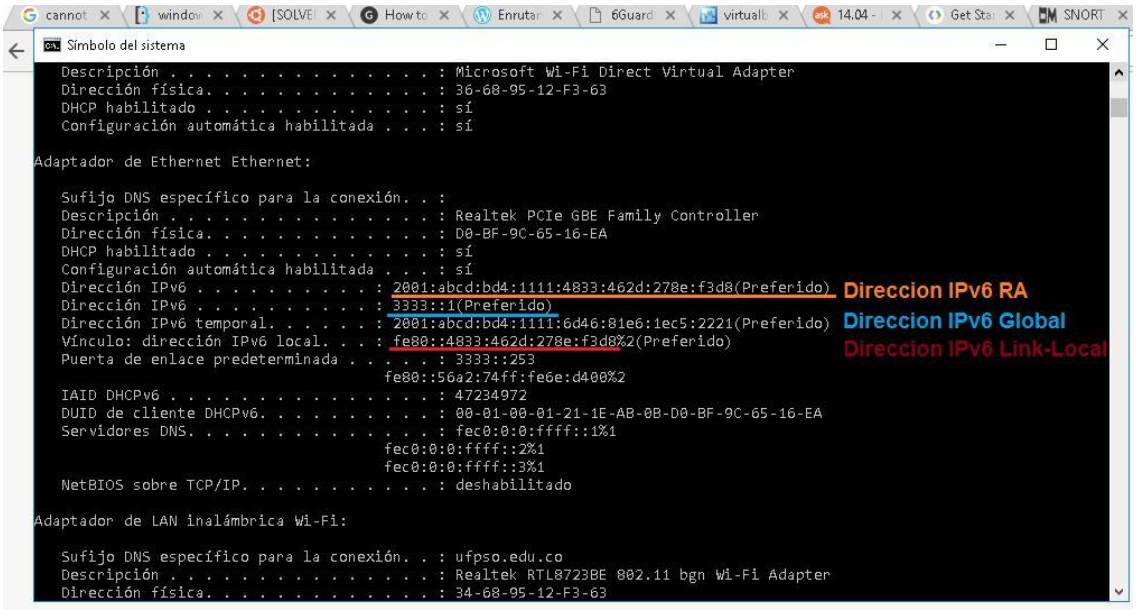

Figura 95. Dirección IPv6 RA del PC Intruso, Fuente: **Autores del Proyecto**

Sabiendo esta información sobre PC el Intruso, se procede a hacer Ping desde este al PC Victima (2222::1/64) y también acceder desde el PC Intruso al puerto 8000 del PC Victima. Ambos casos fueron exitosamente alertados por SNORT el cual protege remotamente a la red 2222::/64.

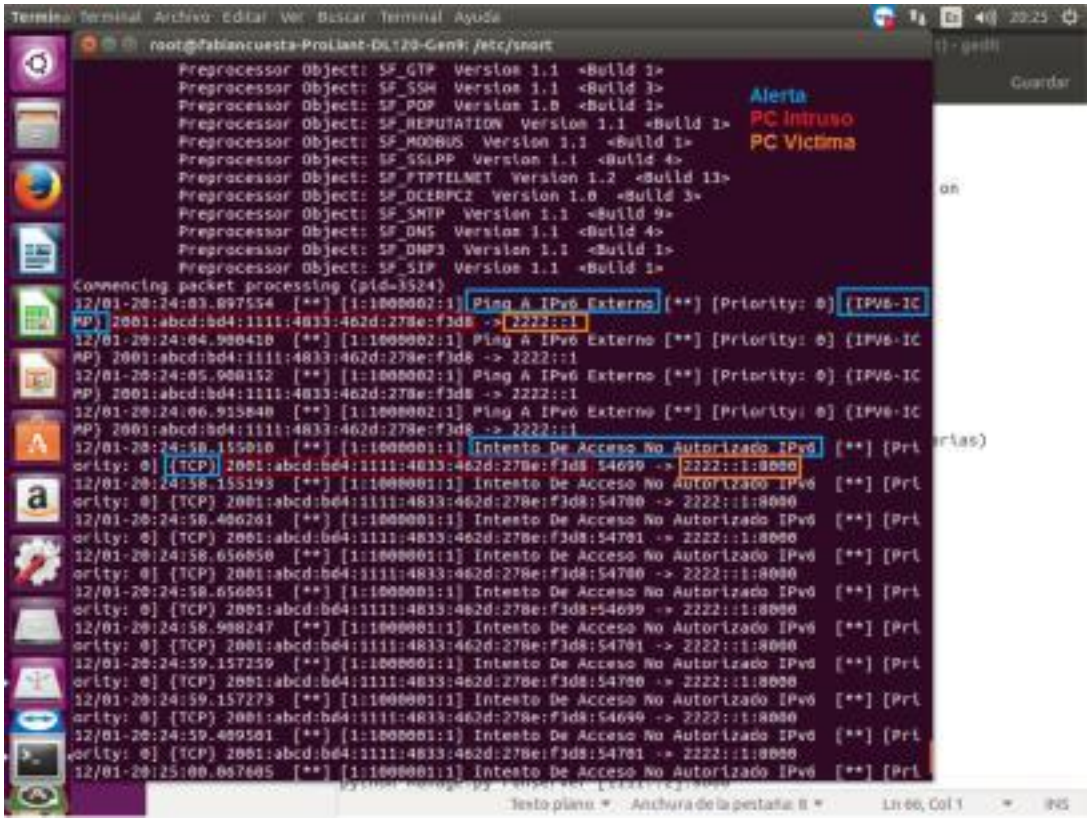

Figura 96. Detección de Alertas ICMPv6 Y TCP remotas Modo Espejo, Fuente: **Autores del Proyecto**

Se confirma así también que la función de **Modo Espejo detiene las alertas de detección de tráfico proveniente del Router**, las cuales son continuas, activando alertas únicamente cuando el PC Intruso (Red Externa) intenta acceder o realizar pings a las direcciones 1111::/64 (local) y/o 2222::/64 (remota) protegidas por SNORT (Red de Hogar).

Con SNORT implementado y configurado bajo IPv6 se procede a realizar la topología final para someter la red Honeynet LAN a los ataques IPv6, con los previamente implementados Honeypots 6Guard (primario), DionaeaFR-IPv6 (secundario) y el ya mencionado sistema detector de Intrusos (IDS) SNORT**. Ahora la Honeynet está lista para ser probada mediante la simulación de ataques de la THC-IPv6 bajo el sistema operativo KALI Linux, y los Logs generados tanto de los Honeypots como del IDS, leídos en un Sniffer de Red.**

Topología LAN Honeynet Final y Prueba De Ataques THC-IPv6

Kali Linux e Instalación de la THC-IPv6

Kali Linux es un proyecto de código abierto mantenido y financiado por Offensive Security, un proveedor de entrenamiento de seguridad de la información de clase mundial y servicios de prueba de penetración. Además de Kali Linux, Offensive Security también mantiene la base de datos Exploit y el curso gratuito en línea, Metasploit Unleashed. (Offence Security)

Kali Linux es una distribución basada en Debian GNU/Linux diseñada principalmente para la auditoría y seguridad informática en general. Trae consigo preinstalados más de 600 programas incluyendo Nmap (un escáner de puertos), Wireshark (un Sniffer), John the Ripper (un crackeador de contraseñas) y la suite Aircrack-ng (software para pruebas de seguridad en redes inalámbricas, todo esto se puede descargar gratuitamente de https://tools.kali.org/)

Figura 97. Kali Linux 2017.3 Instalado en Máquina Virtual, Fuente: **Autores del Proyecto**

Es por esta razón, que se eligió este sistema operativo para realizar las pruebas de ataques IPv6 que los Honeypots deberían detectar. Se Descargó la versión 2017.3 la cual es la última versión disponible hasta la fecha de realización de este proyecto, desde su página oficial ya que como es una distribución de Linux, es de licencia libre y por lo tanto su descarga es gratuita. Se instaló tanto como en una máquina virtual de Virtualbox como **físicamente en un computador portátil**. Posteriormente se procedió a realizar la instalación de actualizaciones mediante los ya conocidos comandos, ***sudo apt-get update, sudo apt-get upgrade*** *y* ***sudo apt-get dist-upgrade*** en la terminal.

The Hackers Choice IPv6 (THC-IPv6) es un conjunto completo de herramientas para atacar las debilidades inherentes del protocolo de internet IPv6 y el protocolo ICMPv6, e incluye una biblioteca de fábrica de paquetes fácil de usar. (The Hackers Choice, 2014)

Su última versión hasta la fecha es la v3.2, Se descargó gratuitamente desde su repositorio en Github oficial https://github.com/vanhauser-thc/thc-IPv6 donde se aclara que su uso es solamente para fines legales y se le aplica a su código la licencia de servidores AGPL. (https://www.thc.org) Para que funcione correctamente es necesario descargar unas librerías esenciales para su correcto uso mediante el comando de terminal: ***sudo apt-get install libpcap-dev libssl-dev.***

Luego se procedió a instalar mediante los comando ***sudo make install***, el cual instala THC-IPv6 en la ruta **usr/local/bin** y también el comando ***make*** para instalarlo en la misma carpeta de descarga como lo muestra la siguiente imagen

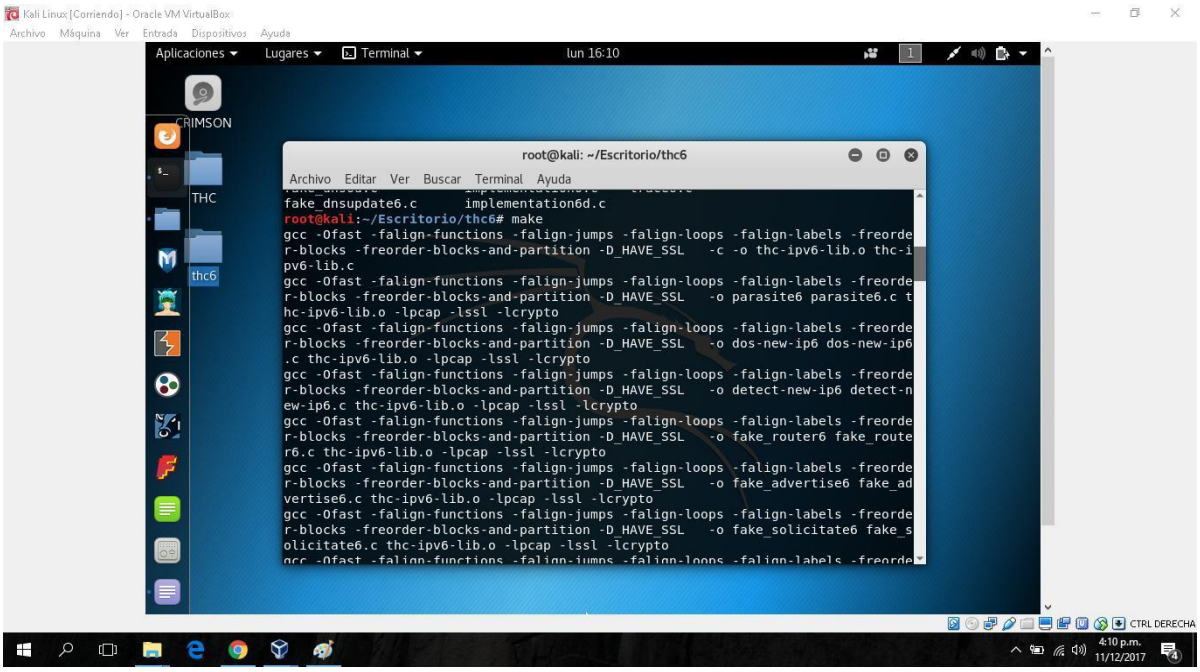

Figura 98. Instalación de la THC-IPv6 en Kali Linux, Fuente: **Autores del Proyecto**

Luego se procedió a verificar que su instalación fue realizada correctamente en la carpeta **usr/local/bin**

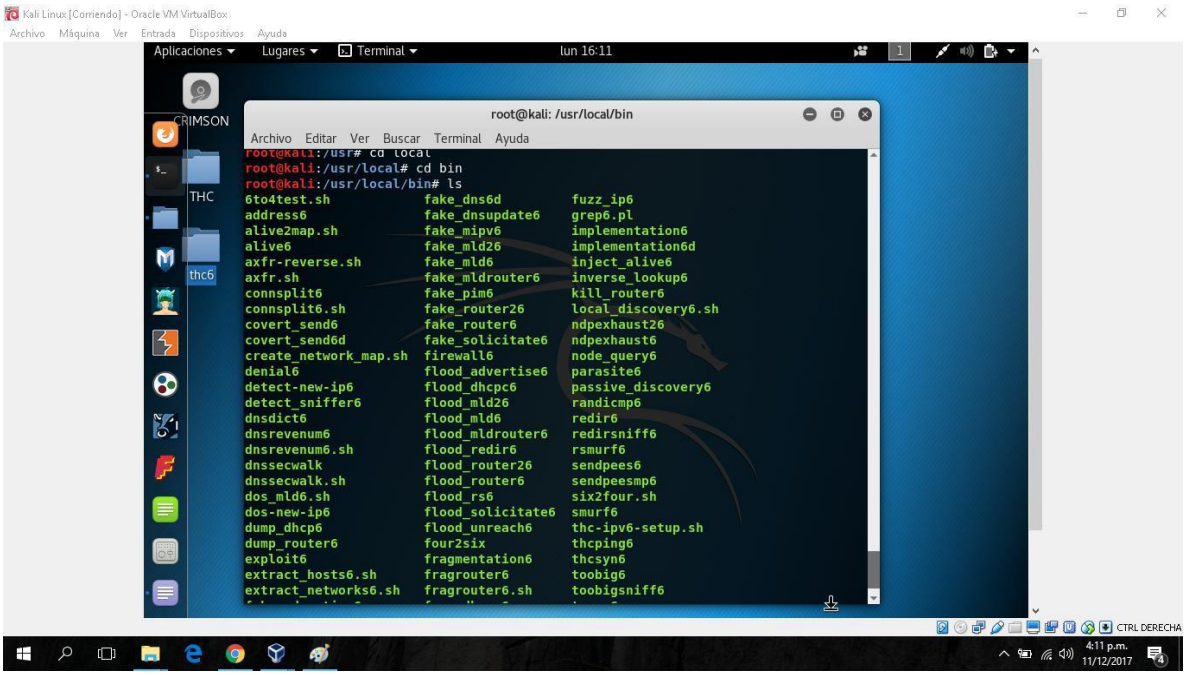

Figura 99. Lista de archivos de THC-IPv6 en Kali Linux, Fuente: **Autores del Proyecto**

El archivo **Readme** con la instrucción de sintaxis de ejecución e información de cada uno de los diferentes ataques de la THC-IPv6 se puede encontrar más detallado en la página de Kali Tools: https://tools.kali.org/information-gathering/thc-IPv6.

Para la Topología Final que se mostrara a continuación se le instalo Kali Linux físicamente a un computador portátil que actuara como *atacante* (PC 1), se le asignó una dirección IPv6 global automática para establecer comunicación directa con el Router.

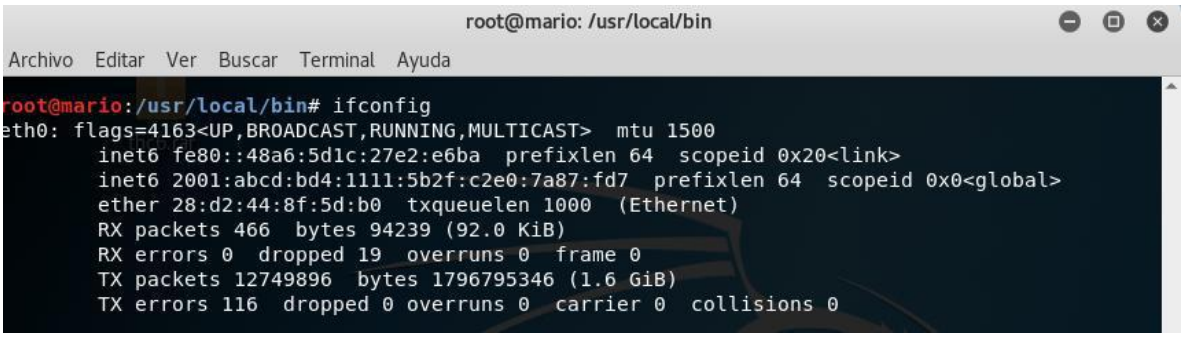

Figura 100. ifconfig de PC 1 Atacante Kali Linux, Fuente: **Autores del Proyecto**

Topología LAN Honeynet Final

La Topología de Honeynet Utilizada para verificar el funcionamiento de los Honeypots, SNORT y de los ataques THC-IPv6 es la siguiente:

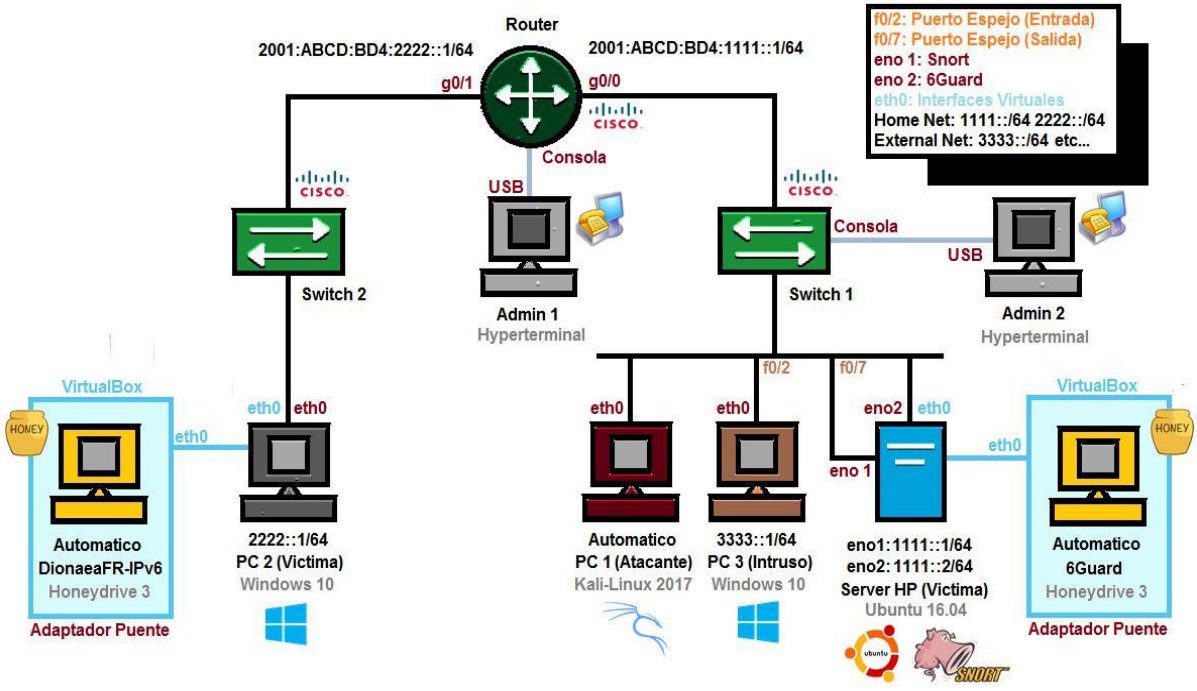

Figura 101. Topología LAN Honeynet Final, Fuente: **Autores del Proyecto**

Las Consideraciones previas son las siguientes:

- El Entorno de la topología Honeynet es LAN y sin ningún tipo de conexión a Internet.

- Está totalmente realizada en IPv6. (Nada de IPv4 o tuneling)

- Es de tipo IPv6 Unicast-Routing.

- Interactúan Los Sistemas Operativos, Windows, Ubuntu, Kali Linux y Honeydrive.

- Los Firewalls de cada equipo están todos desactivados.

- 6Guard es el Honeypot Principal de la Red Honeynet. Debe reconocer todos los ataques realizados por parte de la THC-IPv6 a nivel de enlace-local.

- 6Guard es capaz de reconocer ataques a cada miembro de la red así como también ataques que se hagan a toda la red en sí.

- DionaeaFR-IPv6 es un Honeypot Secundario. Se sabe que su función es principalmente reconocer ataques provenientes de la Internet. Aun así se dejara este Honeypot en la Honeynet para cuando se desee establecer conexión a internet por medio de IPv6.

- Kali Linux 2017.3 se instaló Físicamente en un computador portátil y no en máquina virtual para esta topología.

- Para dejar al PC Atacante con Kali Linux del lado del Switch 2 sería necesario instalar otro 6Guard de ese lado.

- Es necesario conectar los Honeypots de la máquina virtual y el Atacante, al Router por medio de la asignación de IPv6 Automática (Global).

- Al realizar el enrutamiento estático IPv6 con direcciones Globales, automáticamente se generan a cada máquina una dirección de anuncio de Router y una de Enlace-Local.

- Las direcciones de Enlace-Local generadas automáticamente son las que se utilizaran para la realización y detección de ataques IPv6 por parte de 6Guard.

- Se añadió un PC 3 (Intruso) adicional para verificar la funcionalidad de SNORT, Este solo accederá a los servicios FTP, UDP e ICMPv6 de los demás computadores.

- SNORT contiene como Red Local a 1111::/64 (local) y 2222::/64 (remota)

- SNORT contiene como Red Externa todo lo que no sea la Red Local. (El Intruso).

- La Red Local actuara como Victima de Atacantes pero protegida con Honeypots.

- La Interfaz eno1 del Servidor HP cuenta con la realización de Puerto Espejo para la detección garantizada de SNORT (por allí es donde este se comunica), y la eno2 no cuenta con Puerto Espejo ya que por esta es que se comunica 6Guard para verificar su funcionamiento en esa condición, ya que se recomienda la implementación de esta función para aumentar su precisión de detección de ataques.

- Para dejar al PC Intruso del lado del Switch 2 sería necesario hacer VLAN, conectar ambos Switch por modo Trunk y finalmente realizar puerto espejo al puerto por donde ingresa dicha VLAN.

- Un solo PC podría actuar de Atacante y de Intruso a la vez, mediante el uso de Kali Linux como Máquina Virtual.

- Los Switch y Router utilizados son de la marca Cisco con soporte para Puerto Espejo e IPv6 Unicast-Routing respectivamente. Administrados y configurados por medio de Hyperteminal.

- Los archivos de registro de los Honeypots y de SNORT serán revisados con Wireshark.

***Figura 102*.** Topología LAN Honeynet Final en el Laboratorio, Fuente: **Autores del Proyecto**

Prueba de Ataques THC-IPv6

Se realizó la **prueba de ataques THC-IPv6** para probar la funcionalidad de la Honeynet implementada en el Laboratorio. Se escogieron ataques de los cuatro grupos de la THC-IPv6. Todos fueron exitosamente detectados por el Honeypot 6Guard. Pero por el contrario DionaeaFR-IPv6 no detecto actividad alguna, ya que como se mencionó anteriormente este Honeypot es eficaz a la hora de conectar una Honeynet a internet por medio de IPv6.

“En IPv4, se sabe qué es "ARP", aquí en IPv6, es reemplazado por ND expandido como Neighbor Discovery (Descubrimiento de Vecinos). ND combina la funcionalidad de ARP, ICMP, ICMP-Redirect y descubrimiento de Router que está presente en IPv4 (…). Básicamente, hay 5

tipos de mensajes ND: Solicitud de Router, Anuncio de Router, Solicitud de vecinos, Anuncio de vecinos y Redireccionamiento" (Kali Linux Tutorials, 2015)

Prueba de Ataques THC-IPv6 Grupo I

El primer grupo de ataques de la THC-IPv6 (**Grupo I**) son los que realizan anuncios falsos de Router, esto es que el atacante se introduce a la red como un Router de alta prioridad y los computadores víctima se conectan a este y no al Router real. Este Grupo I también realiza ataques para redirigir el protocolo ICMPv6 y realiza ataques DOS si se proporciona una dirección de MAC local o de enlace no existente. Se seleccionó el ataque **fake_router6** perteneciente al Grupo I**.**

"Fake_router6 es una herramienta dentro de las herramientas THC-IPv6 incluidas dentro de Kali Linux para probar las vulnerabilidades de ataque y la complejidad en los protocolos IPv6 e ICMPv6. Aquí podemos enfocarnos en el segundo tipo de mensajes ND, Anuncio de Router. Un Router IPv6 envía paquetes de RA de forma irregular que contienen la información de la capa de enlace a la dirección de multidifusión. Esto puede contener información sobre la dirección de la capa de enlace del Router, el rango de red, MTU, etc. Es requerida para el host. Cuando un servidor o máquina cliente ingresa a la red, recibe este paquete RA y se conecta al Router correspondiente y obtiene una dirección IPv6 definida en el rango. (Kali Linux Tutorials, 2015)"

Se utilizó la siguiente línea de comando para ejecutar este ataque:

fake_router6 eth0 bad::00/64

De la cual *eth0* es la interfaz de salida y *bad::00/64* es una dirección de enlace local que se le añadió a este Router falso creado por fake_router6 al cual se conectan los PCs víctimas, también funciona con direcciones de MAC falsas o con DNS falsos.

Nota: Esta información de Router Falso se puede visualizar en los PCs victimas mediante el comando ipconfig (Windows) o ifconfig (Linux) en cmd o terminal al reiniciar el adaptador Ethernet.

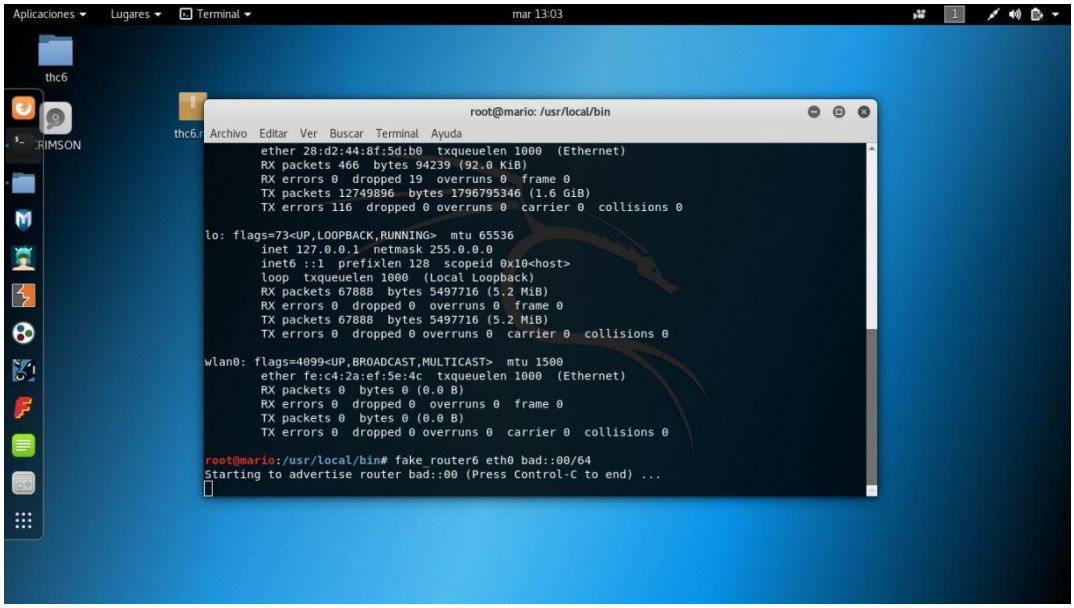

Figura 103. Ejecución de fake_roter6 en Kali Linux, Fuente: **Autores del Proyecto**

Por su parte 6Guard reconoció exitosamente este ataque realizado a toda la red detectando incluso la dirección de enlace local y la MAC del atacante:

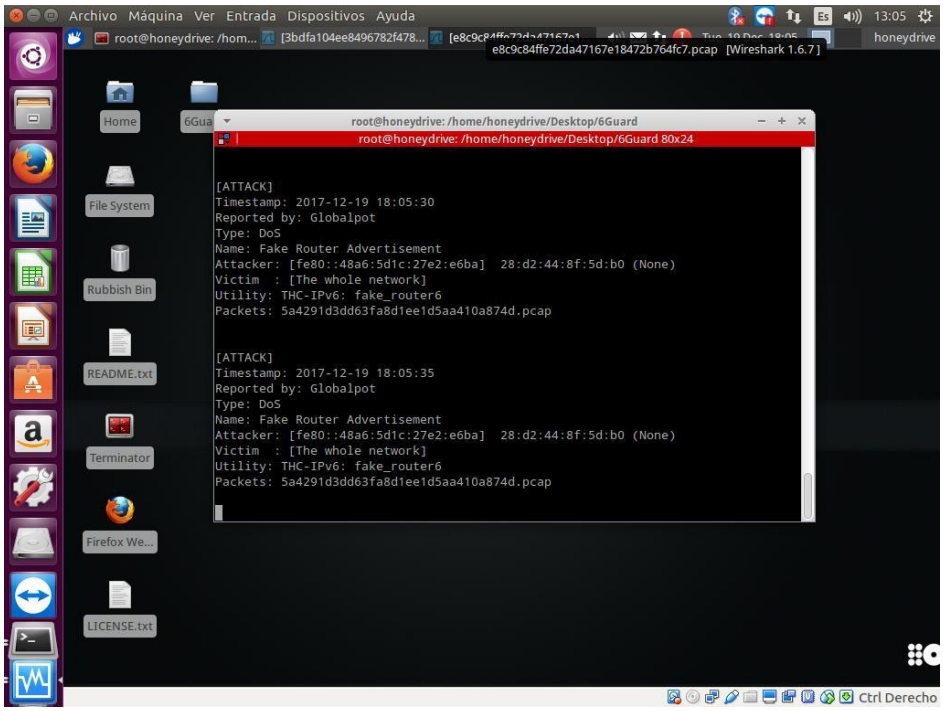

Figura 104. Detección de fake_router6 por 6Guard, Fuente: **Autores del Proyecto**

Prueba de Ataques THC-IPv6 Grupo II

El segundo grupo de ataques de la THC-IPv6 (**Grupo II**) son los que suplantan identidades, direcciones MAC e IP falsas y se anuncian a la red con esta información. También duplican direcciones, impidiendo así que algunos equipos se conecten a la red y desconectando los que ya estaban en la red, también realiza redireccionamiento de tráfico falso y también inundaciones a la red con anuncios y solicitudes falsas. Básicamente cumple con la misma función de los ataques del Grupo I, solo que el Grupo II se enfoca en suplantar equipos y afectar a los demás mientras que el Grupo I se enfoca es en suplantar funciones de Routing. Se seleccionó el ataque **fake_advertise6** perteneciente al Grupo II.

fake_advertise6 cumple la función de anunciar la dirección IPv6 en la red (con su propia MAC si no se especifica), enviándola a la dirección de multidifusión de todos los nodos si no se establece una dirección de destino. La dirección IP de origen es la dirección anunciada si no está

configurada. Como es bien sabido IPv6 no implementa Broadcast, pero sin embargo se puede lograr el mismo efecto mediante el uso de la dirección ff02::1, la cual define grupo de Multicast de enlace-local todos los nodos (all-nodes), así que si se quiere obtener un efecto Broadcast en IPv6, que mejor que enviar paquetes a esta dirección. Ataques como fake_advertise6 pueden enviar paquetes a esta dirección de todos los nodos para afectar una Víctima o incluso toda la red.

Se utilizó la siguiente línea de comando para ejecutar este ataque:

fake_advertise6 eth0 fe80::2c38:68f7:7920:3e7a ff02::1 66:66:66:66:66:66

De la cual *eth0* es la interfaz de salida, *fe80::2c38:68f7:7920:3e7a* es la dirección de enlace local del PC Victima a atacar (Server HP para este caso), *ff02::1* es el blanco de ataque de la red por el cual se permite realizar el ataque especifico y *66:66:66:66:66:66* es una dirección MAC falsa que se le crea al atacante para que se anuncie a la red con esta.

Nota: al momento de crear una MAC o IP duplicada, inmediatamente el PC Victima con la MAC o IP original se queda incomunicado ya que la prioridad la toma el Atacante, esto se puede comprobar fácilmente mediante el uso del comando Ping arrojando tiempo de espera agotado y perdiendo así todos los paquetes cuando anteriormente el equipo estaba comunicado y con envió de paquetes ICMP. (Hogg & Vyncke, IPv6 Security, 2008)

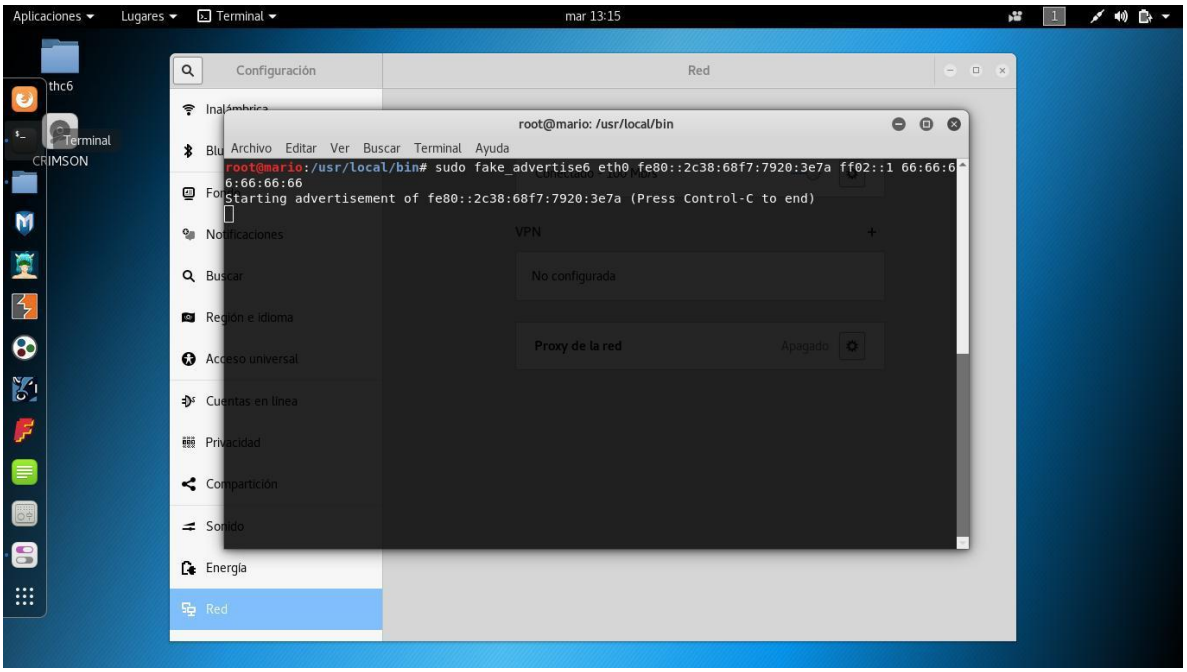

Figura 105. Ejecución de fake_advertise6 en Kali Linux, Fuente: **Autores del Proyecto**

6Guard reconoció exitosamente este ataque detectando que se atacó por medio de la dirección de todos los nodos la dirección de enlace local afectada. La dirección MAC falsa es detectada.

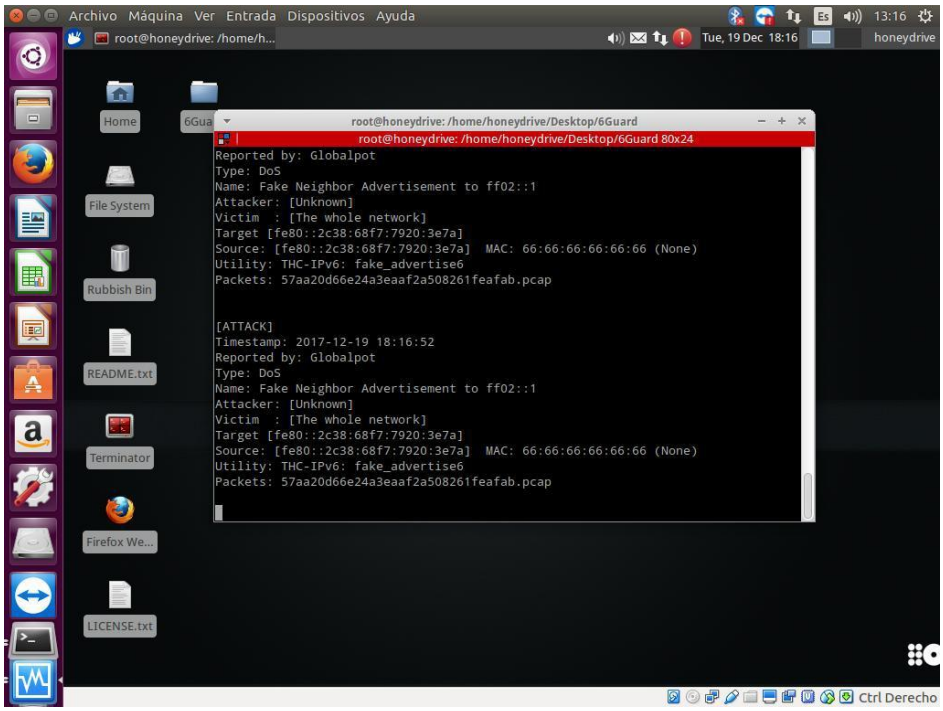

Figura 106. Detección de fake_advertise6 por 6Guard, Fuente: **Autores del Proyecto**

Prueba de Ataques THC-IPv6 Grupo III

El tercer grupo de ataques de la THC-IPv6 (**Grupo III**) son los que proporcionan falsa información de servidor, esto incluye suplantación o falsificación de DHCPv6, DNS o cliente. Los ataques pertenecientes a este grupo usualmente agotan el servicio ofrecido por estos protocolos y trabajan de la mano con ataques pertenecientes al Grupo II. También se usan para afectar todo tipo de funcionamiento relacionado con los protocolos que proporcionan servicios de red, tal como denegaciones e inundaciones. Se seleccionó el ataque **flood_dhcpc6** perteneciente al Grupo III.

flood_dhcpc6 es un ataque que cumple con la función de agotar las direcciones disponibles (que se asignan por medio de DHCPv6) suministradas dinámicamente por un servidor DNS. Como este proyecto está enfocado a un entorno LAN, no se consideró necesario incluir un servidor DNS en la red, sin embargo el ataque fue realizado en el caso que la Honeynet se

conectara a internet para agotar las direcciones automáticas suministradas por *google.com*, es por esto que a la hora de la detección de este ataque por parte del Honeypot 6Guard, no se generaron paquetes de esta detección pero si se generó la alerta correspondiente.

Se utilizó la siguiente línea de comando para ejecutar este ataque:

flood_dhcpc6 –n –l –d google.com eth0

De la cual la opción *–n* usara la MAC real, *-l* solicita una dirección pero no la adquiere, *-d* sirve para forzar actualizaciones DNS, *google.com* es el servidor y *eth0* es la interfaz de salida.

Nota: el comando *–N* usa la dirección link-local aparte de la MAC real, se recomienda ejecutar el ataque **parasite6** (del Grupo II) en paralelo cuando no se ejecuta *–N*, este último ataque es generalmente ejecutado en conjunto con otros ataques ya que es un Spoofer (suplantador) de ARP y redirecciona tráfico falso.

Nota 2: Se puede introducir una dirección IPv6 que suministre el Servicio de Dominio.

Nota3: Si no se introduce *–n* ni *–N*, se tomara una MAC aleatoria falsa.

Nota 4: Si el pool de direcciones que ofrece el servidor DHCPv6 es demasiado grande, no tiene sentido el uso de este ataque solo.

Nota 5: El ataque generalmente termina cuando el atacante corta el proceso, debido a que es inundación de direcciones, se tomara muchos paquetes de los cuales con uno solo de ellos es suficiente para poder ser analizado por un Sniffer de red. Debido a que para este proyecto no se tiene conexión real a *google.com* solo se generó una sola alerta de la solicitud de inundación DHCP por parte de flood_dhcpc6 y de una manera superficial.

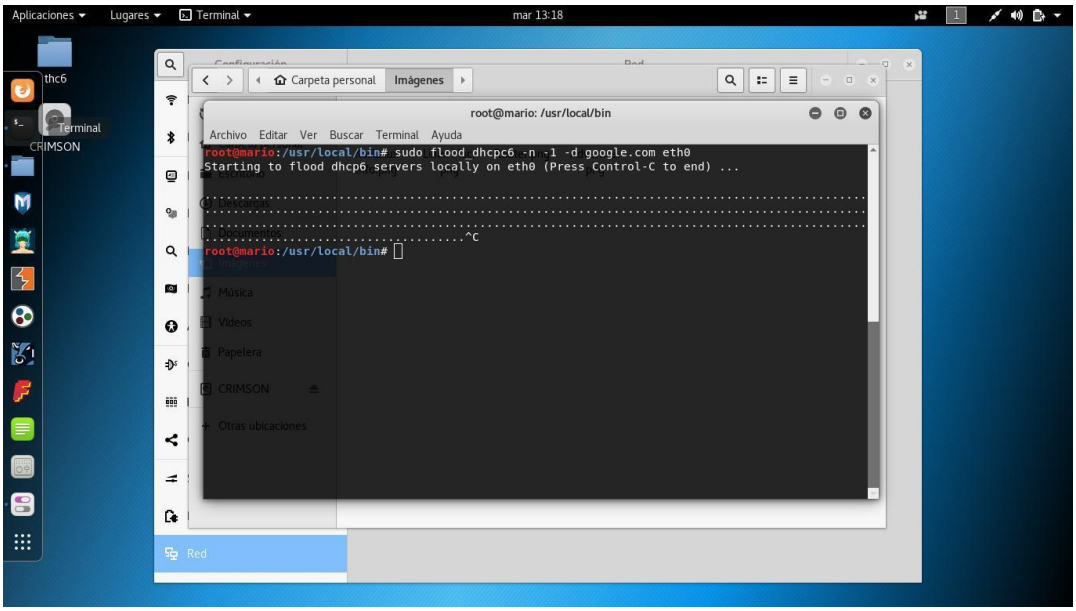

Figura 107. Ejecución de flood_dhcpc6 en Kali Linux, Fuente: **Autores del Proyecto**

6Guard reconoció el ataque pero como no hay conexión a internet no se generaron paquetes. Debido a esto, en la imagen siguiente se generó solo una alerta (la última que aparece).

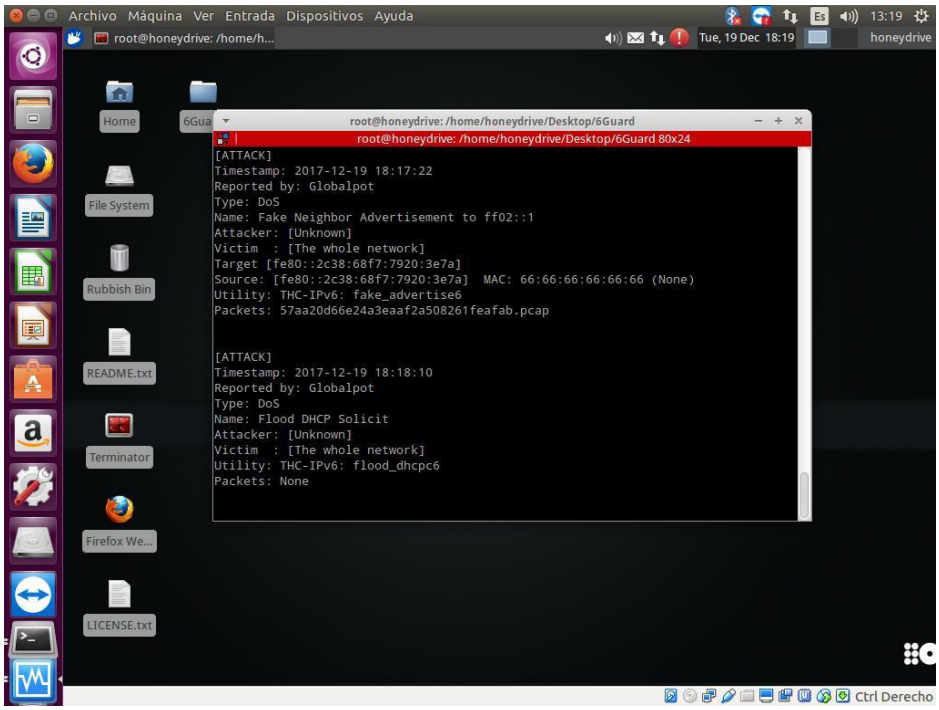

Figura 108. Detección de flood_dhcpc6 por 6Guard, Fuente: **Autores del Proyecto**

Prueba de Ataques THC-IPv6 Grupo IV

El cuarto grupo de ataques de la THC-IPv6 (**Grupo IV**) son los que realizan ataques de consumidor maligno, esto es que consumen mucho recurso de una red o de la CPU de una máquina. Solo sirven para ocasionar daño a un elemento en sí. Este grupo de la THC intenta desde un principio realizar ataques DOS, inundaciones, calentamientos a servidores, ataques pitufos (Smurf) etc… Así que sus principios son diferentes a los de los otros grupos que más que todo se basan en el Spoofing.

A Continuación se cita un comentario de la página http://kalilinuxtutorials.com acerca del escenario realizado para la prueba de uno de los ataques del Grupo IV de la TCH: "Para ser sincero contigo esto es bastante vandálico. Esto bloquea todos los sistemas en la red objetivo y no solo el host víctima. Para este simple tutorial tuve que prepararme mucho porque la realización de este ataque mata todo lo que hay en la red. Así que tuve que moverme a la máquina en vivo para completar este tutorial." (Kali Linux Tutorials, 2015)

Se seleccionó el ataque **sendpeesmp6** perteneciente al grupo IV. Este ataque se encarga de enviar muchísimos mensajes de solicitación de vecino, por lo cual si envía como blanco para realizar los ataques la dirección de todos los nodos (ff02::1) su efectividad incrementaría bastante. Estos mensajes vienen cargados de claves criptográficas, tipo RSA y CGA. sendpeesmp6 es una versión multi-hilo del ataque **sendpees6.**

Se utilizó la siguiente línea de comando para ejecutar este ataque:

sendpeesmp6 eth0 1024 fe80::a00:27ff:fe38:d1ec ff02::1

De la cual *eth0* es la interfaz de salida, *1024* es el parámetro de duración del ataque, *fe80::a00:27ff:fe38:d1ec*, y *ff02::1* es la dirección de blanco de ataque, dirección de todos los nodos.

Nota: Este ataque se realizó de manera rápida, lo suficiente para tomar capturas debido a su alta potencialidad de dañar recursos.

Nota 2: En Windows Se puede visualizar el monitor de recursos para ver el consumo de CPU mientras el ataque es realizado, En Linux por medio del monitor de sistema, es una maniobra un poco peligrosa.

Nota 3: El ataque termina cuando el atacante corta el proceso, debido a que es inundación de mensajes para dañar recursos, se tomara muchos paquetes de los cuales con uno solo de ellos es suficiente para poder ser analizado por un Sniffer de red.

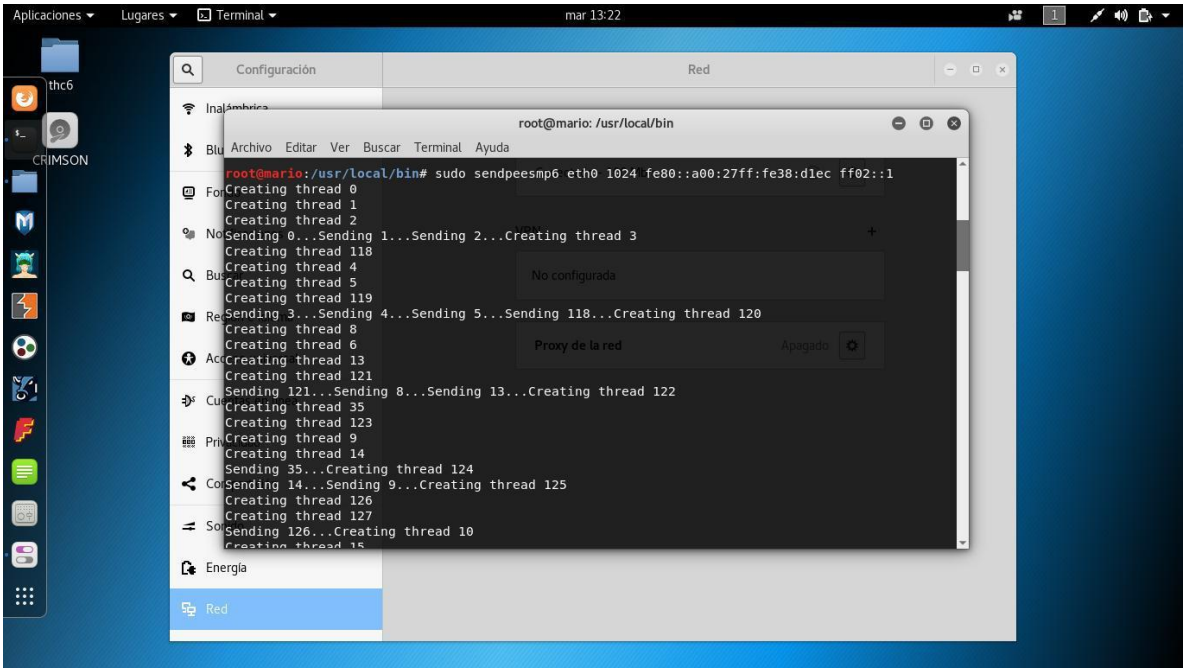

Figura 109. Ejecución de sendpeesmp6 en Kali Linux, Fuente: **Autores del Proyecto**

Se reconoció una gran cantidad de alertas de este ataque, como se ejecutaron varios Honeypots 6Guard cada uno reconoce varios paquetes de esta inundación. Algunos como sendpees6.

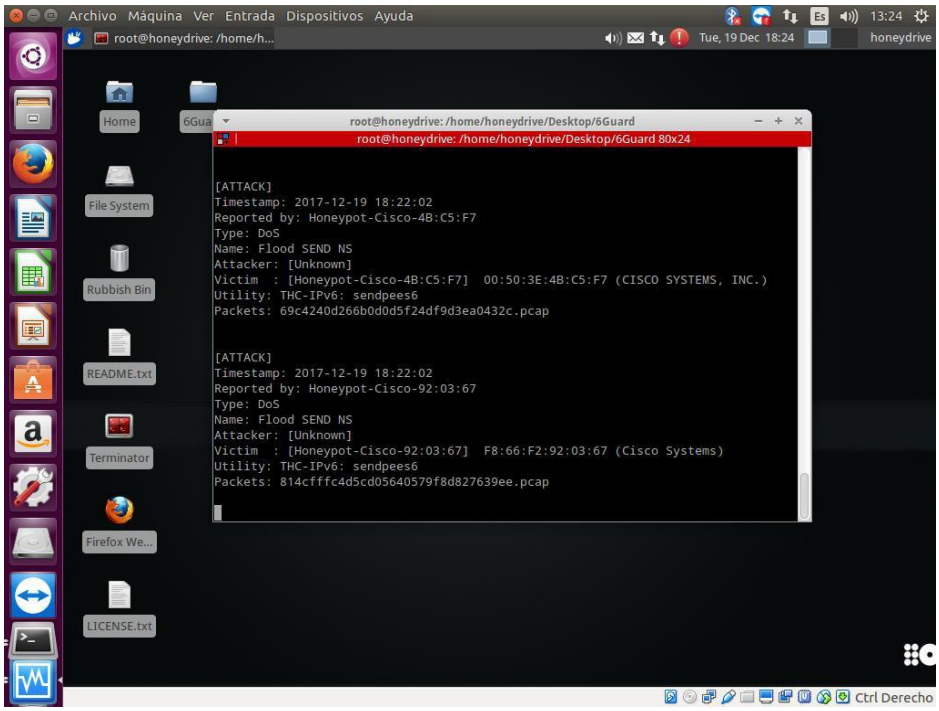

Figura 110. Detección de sendpeesmp6 por 6Guard, Fuente: **Autores del Proyecto**

Imprecisión en Detección de algunos ataques por falta de Puerto Espejo

Como se registró anteriormente en el último ataque sendpeesmp6, 6Guard reconoció algunos ataques como sendpees6. Estas imprecisiones se deben a que la función de puerto espejo no fue utilizada para el puerto eno2 del servidor por el cual es que el Honeypot se comunica con la red y detecta los ataques. Sin embargo esto no es obstáculo para 6Guard en reconocer información de los ataques.

Uno de los ataques más peligrosos del Grupo IV de la THC-IPv6 fue ejecutado cuidadosamente; el ataque **smurf6. "**Smurf6 es una herramienta para realizar un ataque Smurf en la red IPv6. Un ataque Smurf es un tipo de ataque DOS donde un atacante hace sonar la dirección de difusión con una dirección falsa de una víctima. Eventualmente, todos los nodos en la red obtienen una solicitud ICMP ping de la dirección IP de la víctima. Como resultado, todos los hosts responden a la dirección IP de la víctima convirtiéndola en un ataque DDoS. En IPv4, este

ataque no tendrá éxito en la mayoría de los Router y Switch modernos. Pero IPv6 sigue siendo vulnerable.” (Kali Linux Tutorials, 2015)

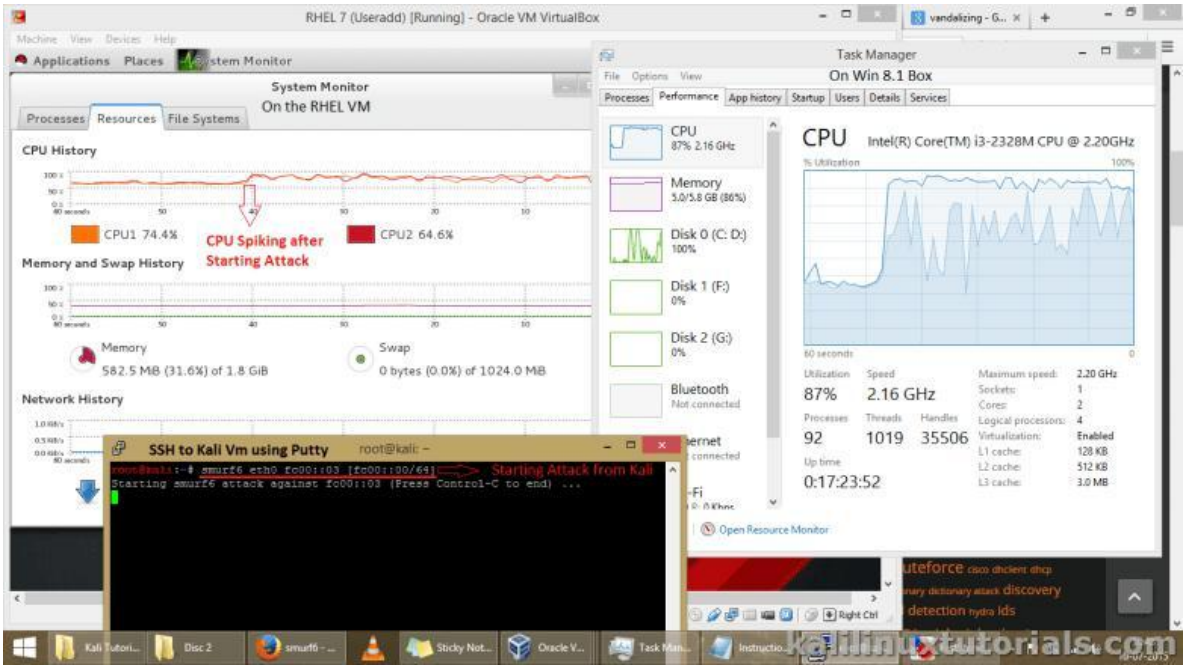

Figura 111. Uso de recurso de CPU elevado en Windows por causa de smurf6, Fuente: **kalilinuxtutorials.com**

Smurf6 es un ataque bastante peligroso y complejo, que funciona perfectamente cuando se quiere inundar toda una red con paquetes ICMPv6. Se recomienda para 6Guard usar la función puerto espejo para reconocerlo totalmente. En este caso 6Guard se ejecutó sin puerto espejo, por lo que al reconocer la alerta de este ataque reconoció solo una parte de Smurf6 y lo reconoció como un ataque **del Grupo de descubrimiento avanzado de Host utilizado por Nmap** pues 6Guard reconoce ataques de los cuatro Grupos de la THC-IPv6 y del grupo Nmap. También reconoció a la víctima como si fuera el atacante.

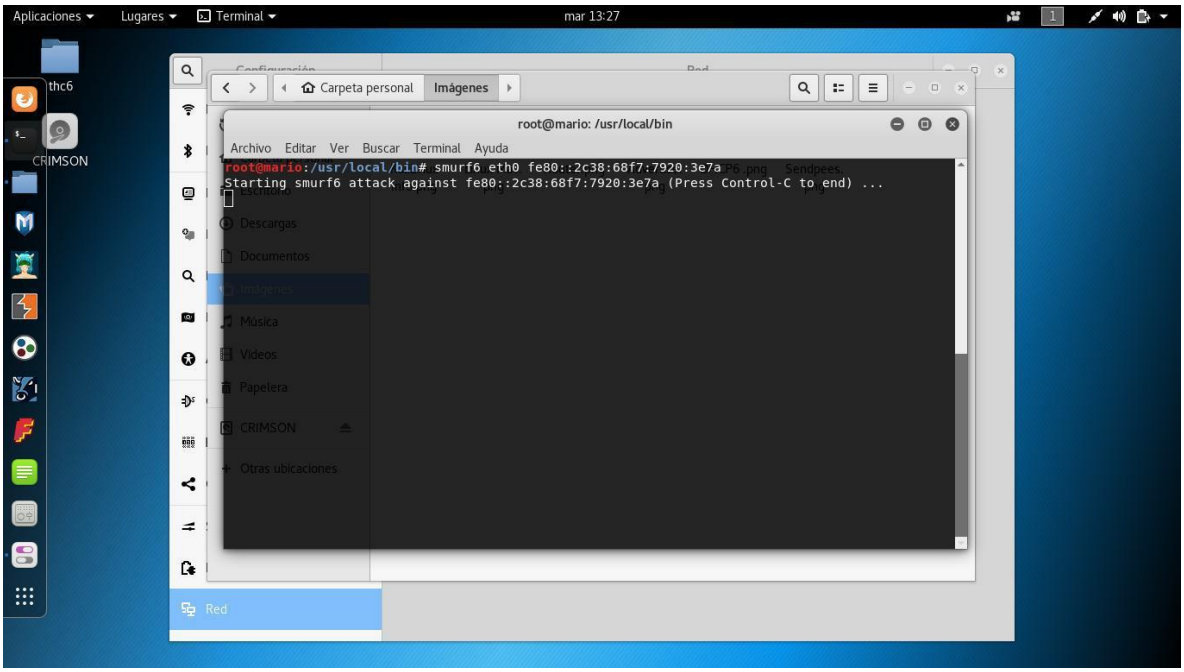

Figura 112. Ejecución de smurf6 en Kali Linux, Fuente: **Autores del Proyecto**

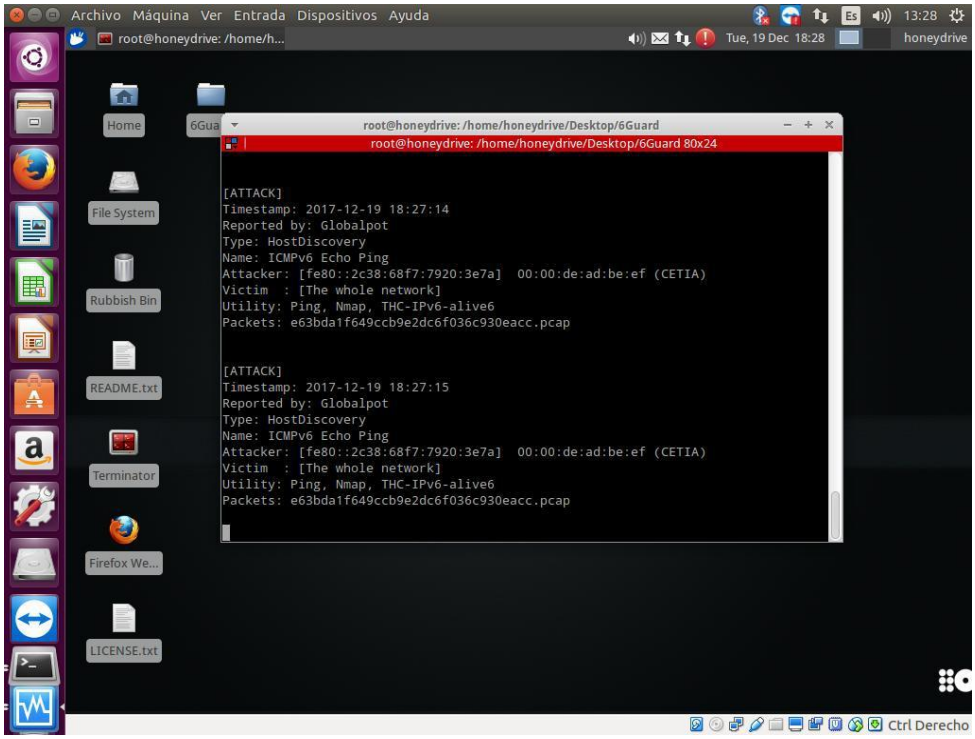

Figura 113. Detección imprecisa de smurf6 por 6Guard, Fuente: **Autores del Proyecto**

Falsas Alarmas

El Sever HP con 6Guard ejecutándose, en un momento fue probado desde el lado del Switch 2, y se conectaba al Router por la interfaz g0/1 de dirección 2001:abcd:bd4:2222::1/64, todo esto para verificar su operatividad. Por error de los ejecutantes del proyecto pero por fortuna para la recolección de información, mientras 6Guard se ejecutaba y se enrutaba por la interfaz g0/1 del Router, fue cambiado el puerto de conexión del Switch2 al Router por la interfaz g0/0 de dirección 2001:abcd:bd4:1111::1/64.

Como la configuración 6Guard había sido realizada ya con el comando ***sudo ./conf_generator.py***, detecto el cambio por la interfaz g0/1 pero a su vez genero una falsa alarma de un ataque tipo fake_router6 proveniente de la interfaz g0/0 del Router, detectando la dirección de enlace local de Router como Atacante (así como también la MAC de esta interfaz) y como Victima a toda la Red.

Esto se explica debido a que como ya se contaba con la configuración de ruta de destino IPv6 la interfaz g0/1 y luego se cambió abruptamente (mientras 6Guard estaba a la escucha sin detenerse aun) por la interfaz g0/0 se generó una inconsistencia de archivos de configuración (carpeta ./conf) y se emitió por error una alerta de ataque del Grupo I que no era más que una Falsa Alarma. A este caso se le puede hacer la analogía de cuando hay dos Antivirus instalados en un mismo equipo y uno de ellos reconoce al otro como un Virus.

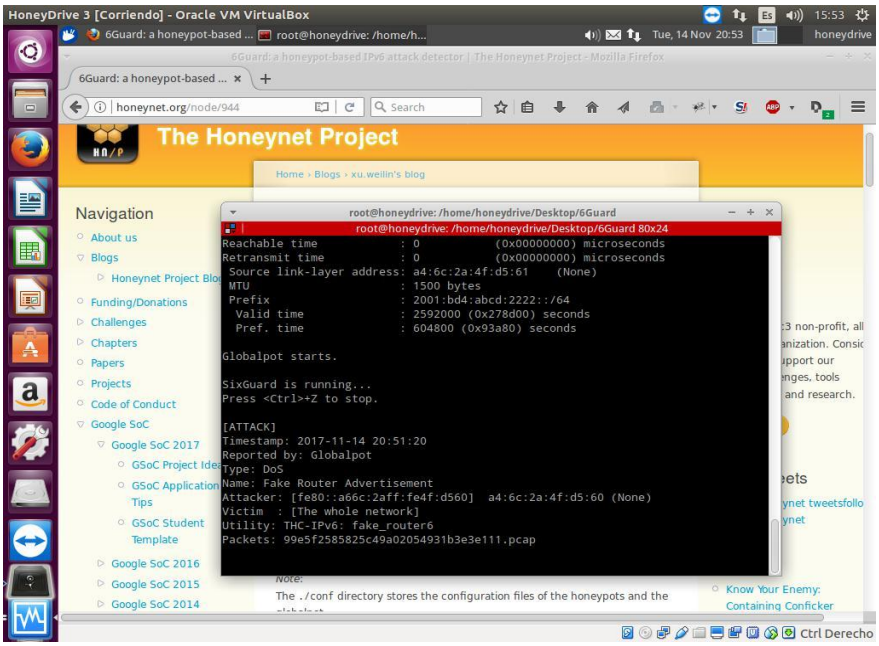

Figura 114. Falsa Alarma generada por 6Guard, reconociendo al Router como atacante erróneamente, Fuente**: Autores del Proyecto**

Resultados de los ataques Realizados

Los resultados de los ataques TCH-IPv6 a nivel de enlace local realizados durante la elaboración del proyecto y detectados por 6Guard fueron resumidos en la siguiente tabla

Tabla 2.

Ataques Realizados Detectados por 6Guard

Ataque de THC-IPv6	**Grupo**	**Detección**
fake_router6	I	Completa
redir6	I	Completa
fake_advertise6	II	Completa
fake_solicitate6	II	Completa
flood_dhcpc6	III	Completa
sendpees6	IV	Completa
sendpeesmp6	IV	Parcial-Completa
smurf6	IV	Incompleta

Fuente: **Autores del Proyecto**

"En el caso de un enfoque más específico de atacantes (es decir, centrarse en víctimas muy específicas), se produciría un problema para detectar un ataque RA / NA falso debido a la ausencia de un puerto espejo de conmutación, a pesar de que el principio de ataque sigue siendo el mismo. El módulo **Globalpot** fue el más activo en la detección de ataques, mientras que solo hubo informes esporádicos del módulo Análisis de eventos. Por otro lado, los informes del módulo de Análisis de Eventos solían ser los más precisos." (Sochor & Zuzcak, 2015)

"6Guard es un detector de ataque IPv6 basado en Honeypot que tiene como objetivo detectar los ataques de nivel local de enlace, especialmente cuando la función de espejo de puerto del Switch no está disponible." (Xu, 2012)

DionaeaFR-IPv6 Por su parte no mostro actividad alguna para los ataques realizados en LAN, la siguiente imagen muestra como DionaeaFR registra ataques provenientes de Internet.

Figura 115. Detección de ataques por DionaeaFR, Fuente: **vanimpe.eu**

Lectura de registros con Wireshark

Wireshark

Toda la información generada por SNORT y por 6Guard (y también por DionaeaFR-IPv6) es almacenada en distintos archivos de formato .log o .pcap que pueden ser leídos manualmente por un Sniffer de red como lo es **Wireshark**.

"Wireshark es un **analizador** de protocolo de red. Le permite capturar y explorar de forma interactiva el tráfico que se ejecuta en una red informática. Tiene un conjunto de características rico y poderoso y es la herramienta más popular de su tipo en el mundo. Se ejecuta en la mayoría de las plataformas informáticas, incluidas Windows, macOS, Linux y UNIX. Los profesionales de redes, expertos en seguridad, desarrolladores y educadores de todo el mundo lo usan regularmente. Está disponible libremente como fuente abierta, y se publica bajo la **Licencia Pública General GNU versión 2**. Es desarrollado y mantenido por un equipo global de expertos en protocolo, y es un ejemplo de tecnología disruptiva." (Wireshark)

Los principales usos que se le puede dar la información recibida por Wireshark son:

- Convierte el tráfico de red en un formato amigable y entendible para los humanos para luego sacar conclusiones de una forma más sencilla.
- Análisis de información específica como fallos, protocolos utilizados, tiempos, direcciones de origen, direcciones destino etc…
- Administra y gestiona la información que pasa a través de una red LAN, en este caso la Honeynet implementada.
- También funciona como sistema de detección de intrusos pero no es específico como lo sería un IDS como SNORT.

- Puede detectar tráfico en tiempo real, como leer archivos provenientes de otras aplicaciones como lo es el formato .pcap. También se pueden generar archivos de registro para otras aplicaciones como Nmap, Nessus, Sebek etc…

- Identificar estabilidad y vulnerabilidades.

Figura 116. Logo de Wireshark, Fuente: **wireshark.org**

Wireshark: Captura a Tiempo real y registros de tráfico SNORT

SNORT también estuvo presente y funcional durante la fase de pruebas de ataques de la THC-IPv6, su función era **alertar accesos no autorizados** a la Red Local (1111::/64 y 2222::/64), provenientes de la Red Externa conformada por el PC intruso con dirección 3333::1/64.

Wireshark viene preinstalado en la mayoría de distribuciones de Linux. En el caso del Server HP con sistema operativo Ubuntu que es donde están instalados SNORT (interfaz eno1) y el Honeypot 6Guard (interfaz eno2), Wireshark ya venía preinstalado y listo para usarse. Para poder acceder correctamente a todas las funciones de Wireshark adecuadamente es necesario ejecutarlo desde la terminal por medio del modo súper usuario root (***sudo wireshark***). Wireshark es capaz de capturar a tiempo real el tráfico proveniente de una interfaz.

La interfaz eno1 fue seleccionada para **capturar el tráfico a tiempo real** detectado también por SNORT con las reglas asignadas:

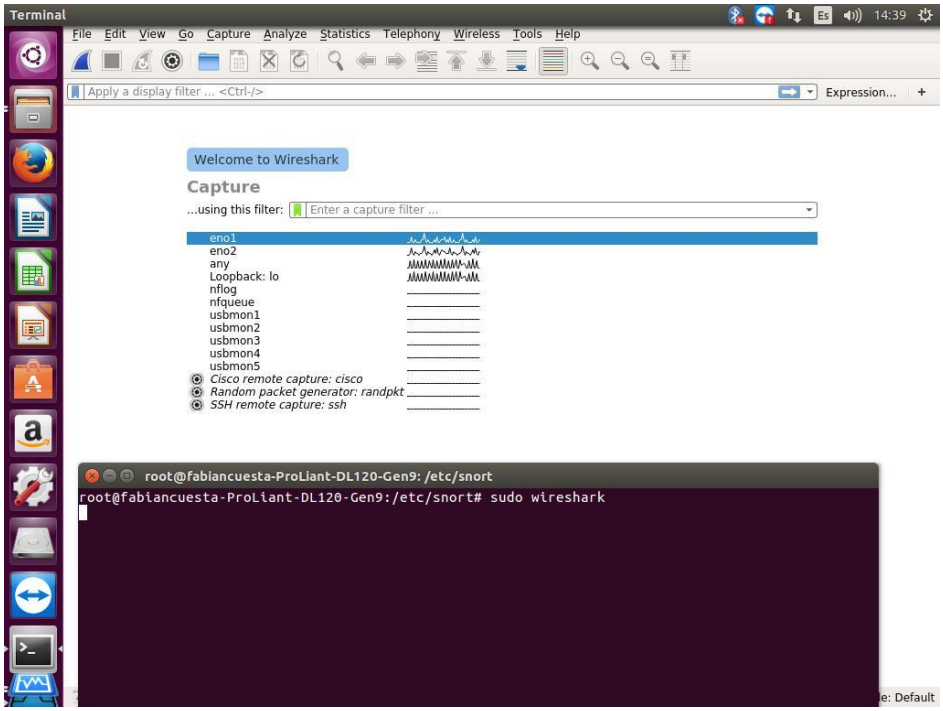

Figura 117. Selección captura de interfaces a tiempo real de Wireshark, Fuente: **Autores del Proyecto**

Cabe Resaltar que SNORT puede generar archivos de registro como **.logs y .pcaps** para que sean leídos posteriormente por Wireshark, así como también Wireshark puede también guardar ese tipo de archivos de registro provenientes de su captura a tiempo real. SNORT solamente se detendrá cuando el administrador de la Red lo desee ya que cuando se pone a la escucha de tráfico su tiempo es indefinido.

De esta forma se tiene registro de toda la información del tráfico detectado a tiempo real proveniente de la interfaz eno1 que cuenta con la función de puerto espejo, información de la

actividad de todo el tráfico IPv6 presente en la LAN Honeynet, TCP, ICMP, UDP etc... También información de la actividad realizada por el PC Intruso que es el principal objetivo para verificar la operatividad de SNORT. **Esta información estará a disposición del administrador de la Red con detalles específicos, fechas y horas exactas.**

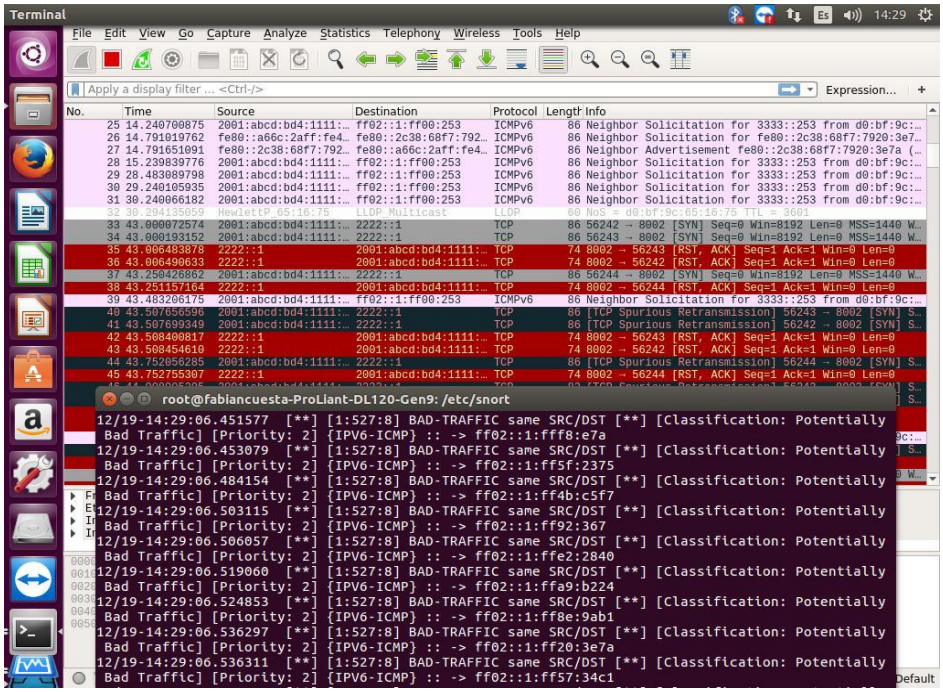

Figura 118. Captura de tráfico a tiempo real de la interfaz eno1 (puerto espejo) por parte de Wireshark y SNORT, Fuente: **Autores del Proyecto**

Wireshark: Lectura de paquetes .pcap de 6Guard y captura a tiempo real de ataques de la THC-IPv6.

La función específica de la Honeynet es detectar y recolectar toda la información disponible del atacante. Así que como se ha dicho anteriormente 6Guard genera paquetes .pcap que se guardan en la carpeta ./pcap, para que sean leídos por el administrador de la red por medio de Wireshark de una forma más concisa y detallada. Luego de finalizada la fase de Ataques de la

THC-IPv6 se cargó uno de los paquetes .pcap generados por la detección de 6Guard de uno de los ataques para visualizarlo en Wireshark. La siguiente imagen evidencia la anterior acción:

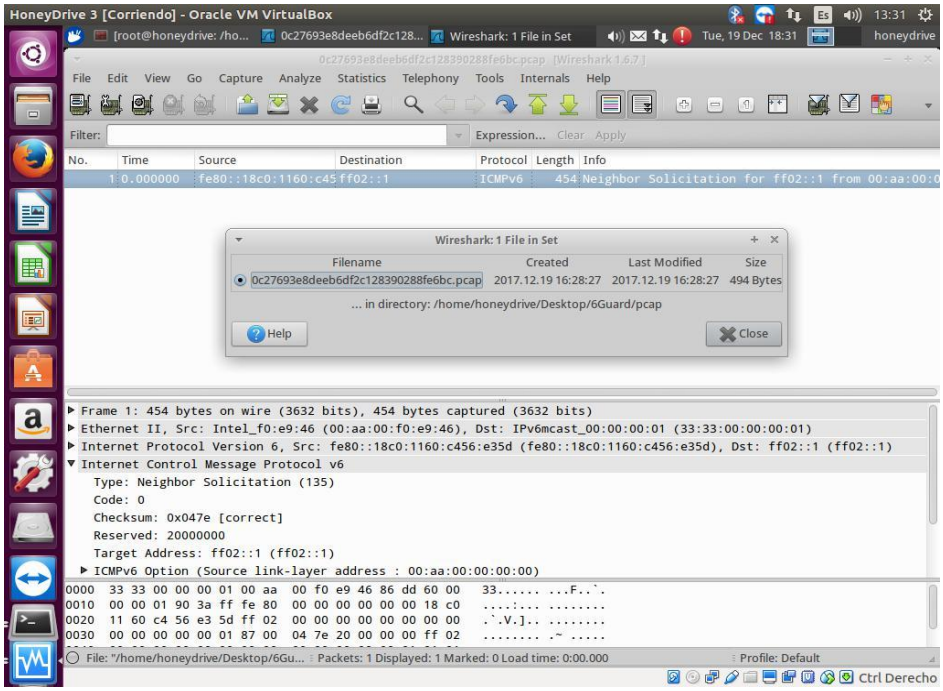

Figura 119. Lectura de archivos de registro .pcap en Wireshark generados por 6Guard, Fuente: **Autores del Proyecto**

Cabe recordar que 6Guard también genera archivos .log (guardados en la carpeta ./log) y estos también pueden ser leídos por Wireshark.

También se realizó captura a tiempo real del tráfico que circulaba por la interfaz eno2 del Server HP, por la cual 6Guard registra los ataques THC-IPv6 realizados en los diferentes lugares de la Honeynet LAN. Wireshark detecta todo el tráfico realizado por los ataques pero sin la detección de que son ataques como tal, simplemente el tráfico recibido por dicha interfaz. Esta información puede ser utilizada para apoyar la suministrada por 6Guard o por cualquier otro tipo de Honeypot.

En la siguiente imagen se puede apreciar el tráfico a tiempo real de la interfaz eno2 capturado por Wireshark. Allí se evidencia la conexión de una dirección IPv6 de enlace local de todos los nodos a un "Router" con dirección de enlace local bad:203… y dirección global 2002:1:2… **Que de no ser por 6Guard no se sabría que toda esa información es falsa y generada por un ataque fake_router6**:

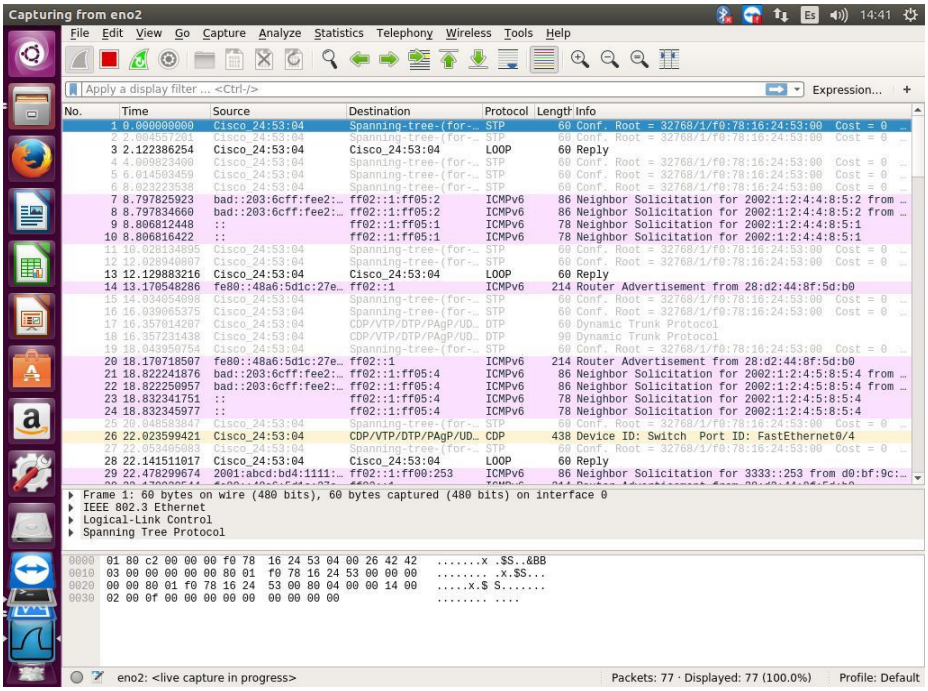

Figura 120. Captura de tráfico a tiempo real de la interfaz eno2 (6Guard) por parte de Wireshark, Fuente: **Autores del Proyecto**

Una vez hecho esto, el administrador de la red humano adquiere todos los datos necesarios del atacante, realiza conclusiones y aprende de su carnada gracias a la recolección y análisis de toda esa información valiosa otorgada por la Honeynet y sus componentes para IPv6 presentados a lo largo de este proyecto. Lo que desee hacer después de esto, puede ser estudiado por otros proyectos de tesis.

Conclusiones

En este trabajo de grado se propuso responder a varios interrogantes que surgen de la transición del IPv4 a IPv6 y los riesgos de seguridad de la información que esto conlleva. Para esto se realiza la implementación de una red Honeynet estática para entornos de red cableada apoyada en IPv6, en el laboratorio de redes y telecomunicaciones de la Universidad Francisco de Paula Santander Ocaña, utilizando herramientas como Ubuntu 16.04 LTS, Oracle VirtualBox, HoneyDrive 3, 6Guard, DionaeaFR, SNORT 2.9.2, Kali Linux 2017.3, Wireshark y las proporcionadas por el kit de THC-IPv6 donde se evidencio un completo conjunto de instrumentos para atacar las debilidades inherentes del protocolo de IPv6 e ICMPv6 y como estos varían en los diferentes sistemas operativos.

Con la finalidad de verificar el comportamiento de los ataques ante la Honeynet implementada se tomaron los detectados por 6guard que son subdivididos en 4 grupos y uno adicional**.** La ejecución y visualización de los ataques a través de la Honeynet se realizó satisfactoriamente excepto en algunos casos en donde para 6Guard, su precisión de detección disminuye. Estas imprecisiones se deben a falta de funciones mínimas como por ejemplo el Puerto Espejo. Sin embargo esto no es obstáculo para el Honeypot en reconocer información de los ataques.

Por otro lado, para CDROM Honeywall Roo 1.4 se concluye que no hay compatibilidad entre este y el protocolo IPv6 o que no está documentada la implementación de este protocolo a un Honeywall Gateway, entonces por lo que se comprobó anteriormente no es posible realizar la implementación de un Honeywall Gateway a una Honeynet IPv6. Fue deber de los autores de este proyecto mantener alejado lo más posible el protocolo IPv4, evadiendo técnicas hibridas como Tunneling (Teredo) o protocolos híbridos como Nat64, ya que en una red hibrida donde

coexistan IPv4 e IPv6, IPv4 gana importancia y prioridad. Pero sin embargo la herramienta más importante de la UTM de Roo 1.4, el IDS SNORT, si cuenta con compatibilidad y documentación en sus versiones más recientes para su uso en IPv6 (y un proyecto propio solo para este protocolo), por lo cual se pudo implementar exitosamente en la Honeynet LAN IPv6 generando alertas muy confiables de intrusos en la red para los protocolos TCP, UDP e ICMPv6.

Enfocando los resultados finales de 6Guard, se puede hablar que a partir de estos se puede obtener información muy valiosa y que tales resultados pueden ser evaluados de una manera totalmente positiva, el módulo Globalpot fue el más activo debido a su cercanía con Multicast Routing. Como se generan cada día ataques de la THC-IPv6, estos se suman a una nueva ola de ataques IPv6, de la cual 6Guard es la primera recomendación para detectarlos. Junto con SNORT y demás Honeypots secundarios (Como Dionaea para la salida de LAN a Internet) se puede considerar como una herramienta clave para la mejora de la seguridad IPv6 en Entornos LAN. Mediante los paquetes leídos por Wireshark se puede obtener un análisis más humano que sirve para tomar medidas luego de detectar al atacante y haber aprendido de él. También sirve el hecho de que el uso de IPv6 aún es reciente y limitado en las topologías LAN y muchos propietarios de Red al no querer realizar un pago justo para seguridad en IPv6 (debido a su escaso uso con respecto a IPv4), en el momento de que uno de estos Ataques se realice y se pierda información o elementos de valor como resultado de estos, lamentara no haber tenido en cuenta a IPv6 pues como se pudo observar, los ataques de la THC-IPv6 son ataques extremadamente complejos y peligrosos pero de muy fácil uso.

Para terminar se sabe que el uso de seguridad en IPv6 para investigaciones científicas y universitarias es cada vez más utilizado pero a su vez también el uso dañino y criminal de ataques IPv6 también aumenta a medida que se aprende más de ellos.

Referencias

(2012). *HONEYPOTS, MONITORIZANDO A LOS ATACANTES.* españa: inteco.

Cybsec Security Sustems. (2013). Obtenido de Cybsec Security Sustems: http://www.cybsec.com

HoneyDrive. (2015). Obtenido de HoneyDrive: https://bruteforcelab.com/honeydrive

Abbasi, F. (2009). *Experiencias con una Honeynet virtual de la Generación III. Obtenido de Experiencias con una Honeynet virtual de la Generación III.*

Arango, J. (2010). *El Aatcante Informatico.*

ARCERT. (2008). *Manual de seguridad de redes, coordinación de Emergencia en redes Teleinformáticas de la Administración Pública Argentina.*

Argueta, A. (2001). *y también existen personas externas que logran violar la seguridad de la red* .

Ávila Mejía, O. (2011). Migración del Protocolo IPv4 a IPv6 . *Contactos 79*, 55-60.

Barker, K. (6 de Junio de 2013). *Las implicaciones de seguridad de IPv6.* Obtenido de Las implicaciones de seguridad de IPv6: https://www.sciencedirect.com/science/article/pii/S1353485813700680

BruteForce lab. (26 de Julio de 2014). *bruteforcelab.com*. Obtenido de bruteforcelab.com.

Citrix System. (2014). *Introducción a Software Defined Networking.*

Convery, S., & Miller, D. (2004). *IPv6 and IPv4 Threat Comparison and BestPractice.*

Creswell, J. W. (1997). *diseño de investigacion con enfoques cualitativos, cuantitativos y mixtos.* fourth.

CYBORG. (2015 de septiembre de 2015). *Flood_Router6 - Inunde.* Obtenido de Flood_Router6 - Inunde: http://cyborg.ztrela.com/flood_router6.php/

David Watson, J. R. (2008). El proyecto Honeynet: herramientas de recopilación de datos, infraestructura, archivos y análisis. *IEEE*, 3-5.

Durdağı, E., & Buldu, A. (2010). *IPV4/IPV6 security and threat comparisons.*

Edgar A, T. A. (2012). Honeynet Virtual Híbrida en el entorno de red de la Universidad Técnica Del Norte de la ciudad de Ibarra. *repositorio utn*, 98-99.

Gáldamez, P. (2013). Seguridad Informática Actualidad TIC.

Gehrke, K. A. (2015). *The Unexplored Impact of IPv6 on Intrusion.*

GitHub. (2016). *vanhauser-thc / thc-ipv6.* Obtenido de vanhauser-thc / thc-ipv6: https://github.com/vanhauser-thc/thc-ipv6

Hacker, E. (10 de noviembre de 2017). *Microsoft no ha reparado la vulnerabilidad de DoS de IPv6 del año en Windows.* Obtenido de Microsoft no ha reparado la vulnerabilidad de DoS de IPv6 del año en Windows: http://www.livehacking.com/tag/flood_router6/

hackerschoice. (2014). *THC 20TH ANNIVERSARY - TAXX.* Obtenido de THC 20TH ANNIVERSARY - TAXX: https://www.thc.org/

Hogg, S. (2008). *IPv6 Security.* Obtenido de IPv6 Security: https://doc.lagout.org/network/IPv6%20Security.pdf

Hogg, S., & Vyncke, E. (2008). *IPv6 Security.*

Honeywall. (17 de agosto de 2005). *Conozca a su enemigo: Honeywall CDROM Roo.* Obtenido de Conozca a su enemigo: Honeywall CDROM Roo: http://old.honeynet.org/papers/cdrom/roo/index.html

IPv6. (2017). *ICMP for IPv6 Redirect.* Obtenido de ICMP for IPv6 Redirect: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipv6_basic/configuration/xe-3s/ip6b-xe-3s-book/ip6-icmp-redir-xe.pdf

KALI. (15 de Febrero de 2014). *THC-IPV6 Descripción del paquete.* Obtenido de THC-IPV6 Descripción del paquete: https://tools.kali.org/information-gathering/thc-ipv6

Kali. (2015). *Kali linux tutorial.* Obtenido de Kali linux tutorial: http://kalilinuxtutorials.com

Kali. (2017). *Acerca de Kali Linux.* Obtenido de Acerca de Kali Linux: https://www.kali.org/about-us/

Kali. (2017). *Kali Linux Penetration Testing Tools.* Obtenido de Kali Linux Penetration Testing Tools: https://tools.kali.org/

Kali Linux Tutorials. (7 de Julio de 2015). *Kali Linux Tutorials.* Obtenido de Kali Linux Tutorials: http://kalilinuxtutorials.com/fake_router6/

KALITOOLS. (15 de febrero de 2014). *THC-IPV6 Descripción del paquete*. Obtenido de THC-IPV6 Descripción del paquete: https://tools.kali.org/information-gathering/thc-ipv6

LinuxParty. (6 de Julio de 2010). *LinuxParty.* Obtenido de LinuxParty: https://www.linux-party.com/index.php/6000-el-sistema-de-deteccion-de-intrusos-snort--windows-y-linux

Miguel, S., Gomez, N., & Bongiovani, P. (2012). Acceso abierto real y potencial a la producción científica de un país. El caso argentino. *El Profesional de la Información*.

Mora, P. A. (2009). Seguridad informatica Honeypots. En P. A. T, *Seguridad informatica Honeypots* (págs. 7-9). ecuador: cybsec-security-systems.

Offence Security. (s.f.). *Kali.org*. Obtenido de Kali.org: https://www.kali.org/

Open Networking Foundation (ONF). (2017). *Open Networking Foundation*. Obtenido de https://www.opennetworking.org/sdn-resources/sdn-definition

Popoviciu, C. (2013). *TechTarget*. Obtenido de http://searchsdn.techtarget.com/feature/IPv6-SDN-When-worlds-collide-in-a-good-way

README.ipv6. (2007). *Preguntas frecuentes sobre Snort.* Obtenido de Preguntas frecuentes sobre Snort: https://www.snort.org/faq/readme-ipv6

Rodriguez, J. (2011). *Seguridad Informática. Qué es la seguridad de la.*

Rouse, M. (2007). *TechTarget*. Obtenido de http://searchtelecom.techtarget.com/definition/switch

Rouse, M. (2010). *Tech Target*. Obtenido de http://searchservervirtualization.techtarget.com/definition/virtual-switch

Rouse, M. (2012). *TechTarget.* Obtenido de http://searchsdn.techtarget.com/definition/SDN-controller-software-defined-networking-controller

Rouse, M. (2013). *TechTarget.* Obtenido de http://searchsdn.techtarget.com/definition/POX

Rouse, M. (2016). *TechTarget*. Obtenido de http://searchnetworking.techtarget.com/definition/router

Rouse, M. (2016). *TechTarget*. Obtenido de http://whatis.techtarget.com/definition/server

Sánchez, D. A. (2015). "Análisis Y Comparación De La Seguridad En Dos Esquemas De Red, Utilizando Los Protocolos IPV4 E IPV6". Ecuador.

Scheiner, B. (15 de Junio de 2001). *Crypto-Gram.* Obtenido de http://schneier.com.

Schütte, M. (18 de Marzo de 2014). *The IPv6 Snort Plugin.* Obtenido de The IPv6 Snort Plugin: https://www.ernw.de/download/20140318_Troopers14_Snort_IPv6.pdf

Sochor, T., & Zuzcak, M. (2015). *Application of Honeypots in IPv6 Networks.*

Spitzner, L. (2003). Honeypots: Catching the Insider Threat . *ScienceDirect*, 3.

The honeynet proyect. (1999). Obtenido de The honeynet proyect: www.honeynet.org

The Honeynet Project. (31 de Mayo de 2006). *Know Your Enemy: Honeynets.* Obtenido de old.honeynet.org: http://old.honeynet.org/papers/honeynet/

The Honeynet Project. (25 de Mayo de 2007). *Roo CDROM User Manual.* Obtenido de Roo CDROM User Manual: http://old.honeynet.org/tools/cdrom/roo/manual/index.html

The Honeynet Project. (2012). *honeynet.org*. Obtenido de honeynet.org: www.honeynet.org

Vermesan, O., & Friess, P. (2013). *Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems.*

Von solms, V. n. (2013). From information security to cyber security. *sciencedirect*, 98-99.

Wireschark. (2016). *Acerca de Wireshark.* Obtenido de Acerca de Wireshark: www.wireshark.org

Wireshark. (s.f.). *Wireshark.* Obtenido de Wireshark: https://www.wireshark.org/

Wright, C. S. (2011). *IPv6: The End of Security As We Know It.*

Xia, W., Tsou, T., R.López, D., Sun, Q., Lu, F., & Haiyong, X. (2013). *A software defined approach to unified IPv6 transition.*

Xu, W. (2012). *6Guard (IPv6 Attack Detector).*

Yu-Chin Cheng, J. (2015). *Honeynet Concepts and Honeywal Installation.* Obtenido de Honeywall roo 1: https://es.slideshare.net/YuChinCheng/honeywall-roo-1