UNIVERSIDAD NACIONAL DE CHIMBORAZO

FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA EN SISTEMAS Y COMPUTACIÓN

Proyecto de Investigación previo a la obtención del título de Ingeniero en Sistemas y Computación

TRABAJO DE TITULACIÓN

MODELO PARA LA DETECCIÓN DE ATAQUES DE DDoS EN SERVIDORES DE NOMBRES DE DOMINIOS SOBRE UN ENTORNO DE SIMULACIÓN EN LA RED DE LA UNIVERSIDAD NACIONAL DE CHIMBORAZO 2018.

Autor(es):

Dany Xavier Bonifaz Herrera

Marlon Marcelo Miranda Martínez

Tutor:

Ing. Lorena Paulina Molina Valdiviezo., Ph.D.

Riobamba – Ecuador

2018

Los miembros del Tribunal de Graduación del proyecto de investigación titulado: "**Modelo para la detección de ataques de DDoS en servidores de nombres de dominios sobre un entorno de simulación en la red de la Universidad Nacional de Chimborazo 2018**".

Presentado por: Dany Xavier Bonifaz Herrera, Marlon Marcelo Miranda Martínez y dirigida por: Ing. Lorena Paulina Molina Valdiviezo., Ph.D.

Una vez escuchada la defensa oral y revisado el informe final del proyecto de investigación con fines de graduación escrito en la cual se ha constatado el cumplimiento de las observaciones realizadas, remite la presente para uso y custodia en la biblioteca de la Facultad de Ingeniería de la UNACH.

Para constancia de lo expuesto firman:

Ing. Lorena Paulina Molina Valdiviezo
Tutora del Proyecto **Firma**

Ing. Luis Gonzalo Allauca Peñafiel
Miembro del Tribunal **Firma**

Ing. Luis Patricio Tello Oquendo
Miembro del Tribunal **Firma**

AUTORÍA DE LA INVESTIGACIÓN

"La responsabilidad del contenido de este Proyecto de Graduación, corresponde exclusivamente a: Dany Xavier Bonifaz Herrera y Marlon Marcelo Miranda Martínez con la dirección de la Ing. Lorena Paulina Molina Valdiviezo, PhD. y el patrimonio intelectual de la misma a la Universidad Nacional de Chimborazo".

Dany Xavier Bonifaz Herrera
060495715-9

Marlon Marcelo Miranda Martínez
060474884-8

AGRADECIMIENTOS

En el presente proyecto de investigación quiero agradecer primero a Dios por brindarme la salud, las fuerzas, el ánimo, y sobre todo por cumplir este anhelado sueño.

Extiendo mi agradecimiento a las autoridades y maestros de la Universidad Nacional de Chimborazo, quienes me abrieron las puertas para culminar con éxito esta gran etapa de mi vida, agradecer también a mis padres, hermanos que fueron el apoyo y la fuerza para cumplir con mi meta, y por último agradecer a mi compañero de tesis por su trabajo, apoyo y amistad conjuntamente con mi Directora de Tesis que nos apoyó en todo momento.

Dany Xavier Bonifaz Herrera.

En primer lugar, agradezco a **Dios** por haberme brindado protección y fortaleza para superar los obstáculos que se me han presentado en la vida, a mis padres **José Miranda** y **María Susana Martínez** que como personas ejemplares me han enseñado a no rendirme jamás y sobre todo a perseverar para cumplir con mis sueños.

Un infinito agradecimiento a mi directora de tesis, Ing. Lorena Molina por su esfuerzo, paciencia y dedicación, quien con sus conocimientos y experiencia ha hecho posible el poder culminar con nuestros estudios. También me gustaría agradecer a todos los docentes durante mi carrera por impartir sus conocimientos para formarme como profesional y humano.

Marlon Marcelo Miranda Martínez.

DEDICATORIA

Dedicamos este trabajo principalmente a Dios por habernos dado la vida y permitirnos llegar a este punto tan importante en nuestra formación profesional. También dedicamos este proyecto de investigación a nuestras familias y seres queridos, quienes son los pilares fundamentales en cada logro de nuestras vidas, a los docentes de la Carrera de Ingeniería en Sistemas y Computación, quienes nos han apoyado para convertirse en profesionales de calidad, a nuestro tutor de tesis, la Ing. Lorena Molina por brindarnos su apoyo con sus conocimientos obtenidos en su vida profesional para el desarrollo de este proyecto de investigación.

Dany Bonifaz, Marlon Miranda.

ÍNDICE GENERAL

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

RESUMEN

En la actualidad, la seguridad de los sistemas informáticos ha aumentado conjuntamente con la evolución y avance de la tecnología, siendo la protección de datos e información un factor muy relevante para el desarrollo de una empresa. Los ataques Distribuidos de Denegación de Servicios (DDoS) interrumpen los servicios a través del consumo excesivo de los recursos en el servidor. La presente investigación tuvo como objetivo el desarrollo de un modelo de seguridad para la detección de ataques DDoS en servidores de nombres de dominios (DNS) sobre un entorno de simulación, para determinar qué mecanismos son apropiados para mitigar dicho ataque sobre la red de la UNACH y el desarrollo de una guía para su aplicación sobre un entorno real.

La metodología utilizada en esta investigación es de estudio longitudinal, pues se obtuvo datos en distintos momentos durante un período determinado, con la finalidad de examinar sus variaciones en el tiempo. Para ello, se analizó e identificó parámetros de medición como: la cantidad de paquetes procesados por día, el nivel de vulnerabilidad por ataques DDoS y el tiempo de respuesta del servidor.

Para la comprobación de la hipótesis se utilizó la prueba estadística T-Student para muestras relacionadas. De acuerdo con los resultados obtenidos gracias a nuestro modelo para la detección de ataques DDoS y en base a las vulnerabilidades detectadas en la red de la Universidad Nacional de Chimborazo, se concluye que se podrá mejorar la seguridad en al menos un 65%.

Palabras clave: Ataques Informáticos, DDoS, UDP Flood, Seguridad Informática.

ABSTRACT

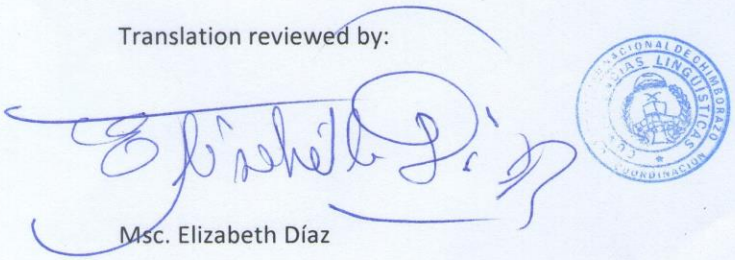
At present, the security of computer systems has increased in conjunction with the evolution and advancement of technology, where data and information protection is very relevant to the development of a company. Distributed denial-of-service (DDoS) attacks disrupt services through excessive resource consumption on the server. This research aimed to develop a security model for the detection of DDoS attacks on domain name servers on a simulation environment, to determine which mechanisms are appropriate to mitigate this attack on a network of The UNACH and development of a guide for its application on a real environment.

The methodology used in this research is a longitudinal study since data were obtained at different times during a given period, in order to examine their variations over time. For which, measurement parameters were analyzed and identified as: The number of packages processed per day, the level of vulnerability by DDoS attacks and the response time of the server.

For the verification of the hypothesis, the T-STUDENT statistical test was used for related samples, for this reason, according to the results obtained thanks to our model for the detection of DDoS attacks; Based on the vulnerabilities detected in the network of the National University of Chimborazo, it is concluded that security can be improved by at least 65%.

Keywords: Computer attacks, DDoS, UDP Flood, Computer security.

Translation reviewed by:



Msc. Elizabeth Díaz

CAPÍTULO I

Introducción

La falta de seguridad es uno de los problemas más trascendentales que influyen en las redes y sistemas de comunicación, donde los ataques informáticos hacen uso de las vulnerabilidades en software y hardware incluso de sus componentes previamente conectados dando un efecto negativo en la seguridad de los mismos afectando incluso a los activos que constituyen la organización.

Existen algunos tipos de ataques que son transmitidos en la red; entre los ataques más comunes se tiene: buffer overflow, shellcodebackdoor, prot standing, sniffing, keylogging, spoofing, trojan, denial of service, Distributed Denial of Service (DDoS) (Molina Lorena, 2015).

En el año de 1999, la Capacidad de Asesoramiento sobre Incidentes Informáticos (CIAC), reportó el primer ataque DDoS (Saman Zargar, J, & D, 2013). Los ataques DDoS impiden que los usuarios legítimos accedan a un servicio específico de la red por medio del envío de peticiones masivas al servidor de la red con el fin de saturar los servicios que este ofrece. Como ejemplo de este tipo de ataque tenemos los siguientes: smurf, fraggle, syn DDoS (Juliette Dromard, 2017).

En la mayoría de los casos, los atacantes reclutan cientos o miles de computadoras las cuales se les considera como ZOMBIE (computador infectado por algún código malicioso) para crear redes llamadas BOTNETS (conjunto de computadoras Zombies). A través de estas redes de computadores se genera tráfico ilegítimo para saturar de peticiones a los servidores web (Cooke, Jahanian, & McPherson, 2014).

Las redes de datos y servicios en Instituciones de Educación Superior son propensas y vulnerables a diversos ataques informáticos. Es por ello que esta tesis busca un mejoramiento y fortalecimiento ante este problema que puede presentarse de diferente manera, ya sea como código malicioso, virus o DDoS que es el caso de este estudio. Una medida muy oportuna es prevenir/detectar un posible ataque de esta naturaleza, por lo tanto, es necesario modelar y simular este tipo de ataque en particular para tener un registro de cómo actúan estos ataques

en la red de la Universidad Nacional de Chimborazo (UNACH) u otro centro de Educación Superior.

En este contexto, la UNACH busca tener un análisis de un ataque DDoS y saber el comportamiento de la red bajo este tipo de ataque para poder contrarrestarlo antes de que los servicios sean denegados por los atacantes.

Planteamiento del Problema

En los últimos años, la seguridad de los sistemas informáticos ha aumentado en igual proporción al avance tecnológico, convirtiéndose en un factor base en el desarrollo económico y social de entidades públicas o privadas. Por tal motivo, el desarrollo y análisis de modelos de seguridad es de vital importancia e indispensable en la protección de los datos. Los datos se los pueden localizar en cualquier ámbito o área, donde las computadoras son una fuente de almacenamiento eficiente en la manipulación de información, además llegando a ser un punto clave para los atacantes que tengan como objetivo a entidades con mayor flujo de datos.

En la actualidad, los ataques de DDoS son una de las formas más comunes para los ciberdelincuentes. Estos suelen planificar el momento del ataque para maximizar el daño mediante métodos de hacking que se inician mediante la propagación de un malware a través de emails, descargas online, servidores, envió de paquetes o datos. Con esto los hackers establecen el control remoto de los servidores y los dispositivos infectados, permitiéndoles dirigir una enorme y continua cantidad de tráfico hacia estos dispositivos hasta que colapsan.

La red de la UNACH es propensa a sufrir ataques informáticos poniendo en riesgo la integridad de la información que se transmite. Por tal motivo, el principal objetivo de la investigación es desarrollar un modelo de seguridad para la detección de ataques de DDoS en servidores de nombres de dominios sobre un entorno simulado, el cual permita en un momento dado replicarse e implementarse sobre la red de la UNACH para mejorar su seguridad aplicando mecanismos de detección de ataques.

Objetivos

Objetivo general

- Desarrollar un modelo para la detección de ataques de DDoS en servidores de nombres de dominios sobre un entorno de simulación en la red de la Universidad Nacional de Chimborazo 2018.

Objetivos Específicos

- Analizar los ataques de DDoS en los servidores de nombres de dominios, y los mecanismos de detección.
- Implementar la infraestructura de red de la UNACH en un entorno de simulación para analizar el comportamiento del ataque DDoS y su mecanismo de detección.
- Desarrollar el modelo de detección de ataques DDoS en la red de la UNACH sobre un entorno de simulación.
- Generar una guía para la aplicación del modelo de detección de ataques de DDoS en un entorno real.

CAPÍTULO II

2. Fundamentación Teórica

Tomando en consideración el conocimiento sobre lo que sucede en la actualidad sobre los ataques DDoS, mecanismos de detección, entre otros, se contextualiza lo siguiente:

¿Qué es un ataque?

Según (Narváez, Romero, & Núñez, 2010) un ataque a la seguridad de red produce un acceso no autorizado, denegando el sistema a través de estas anomalías que acechan en la actualidad, por lo que existen diversas categorías de ataques o amenazas, que son:

2.1. Categorías de ataques o amenazas.

- **Ataques por Interrupción**: Un elemento del sistema es eliminado o se vuelve inutilizado, este tipo de ataque es en contra de la disponibilidad.
- **Ataques por Intercepción**: Cuando un individuo consigue acceso al sistema de manera no autorizada, este tipo de ataque es considerado contra la confidencialidad.
- **Ataques por Modificación**: Cuando un individuo consigue asistir al sistema no sólo de manera no autorizada, sino que puede alterarlo dando así el tipo de ataque contra la integridad.
- **Ataques por Falsificación (Phishing):** Cuando un individuo accede de manera no autorizada e ingresa objetos falsificados dentro del sistema, generando el ataque contra la autenticidad.

2.1.1. El Atacante

El atacante es aquella persona que detecta los puntos más vulnerables de la red y que permiten el acceso al sistema con la finalidad de inundar de paquetes a un determinado servicio a través de una gran cantidad de máquinas (Zombies). Esto tiene como consecuencia el rendimiento pausado, hasta que el servicio quede inutilizado por un determinado tiempo causando incluso grandes pérdidas económicas en la empresa (Hoque.N D. B., 2016).

2.2. Ataques de Denegación de Servicios Distribuidos (DDoS):

Esta técnica de ataque apareció por primera vez en junio de 1998 y actualmente es la técnica más eficiente y difícil de detectar por su naturaleza distribuida.

La Denegación de Servicio Distribuido (DDoS) es considerado un ataque peligroso que infecta de gran cantidad de peticiones ficticias a un determinado servicio de la red. Con esto logra que el servicio se detenga, generando una sobrecarga en la utilización del mismo y, por lo tanto, un incremento exponencial de anomalías (J.Mirkovic & Reiher, 2014), (J.Arzamendia & F.Lopez, 2016).

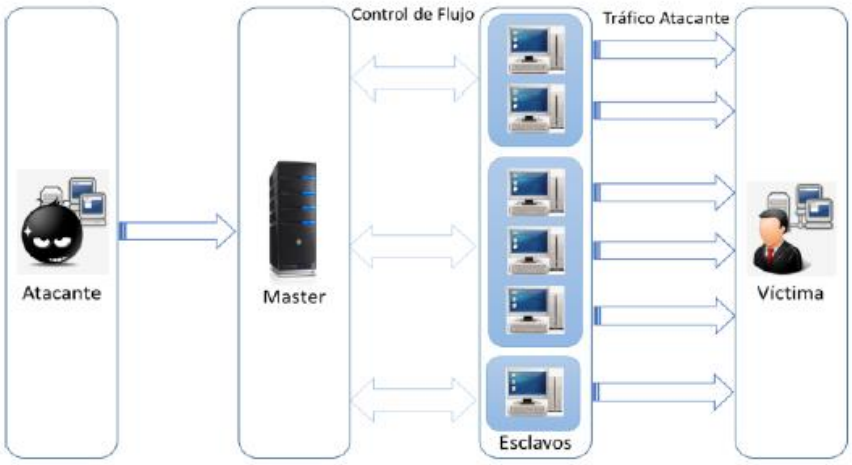


Figura 1. *Arquitectura de un ataque DDoS (Molina Lorena, 2015).*

Los ataques DDoS son los más frecuentes y aumentan a diario debido al inmenso desarrollo de redes informáticas y en conjunto a varias aplicaciones que causan daño a las redes y sistemas de información. Por ejemplo, los botnets son una amenaza crítica que tienen como consecuencia disminuir el ancho de banda y los recursos de los sistemas (Hoque.N, Dhruba.K, Bhattacharyya, & Jugal.K, 2015).

Los ataques DDoS pueden ser clasificados de acuerdo a su dimensión. Los ataques basados en la taxonomía se clasifican en: grado de automatización, exploración de vulnerabilidades, tasa de ataque dinámico y según su impacto (Lau, Stuart, & Michael , 2012).

Un tipo de ataque según dicha clasificación es **UDP Flood**. Un ataque de esta naturaleza es posible cuando el atacante envía un gran volumen de paquetes IP con datagramas UDP a un puerto aleatorio de la víctima. El envío excesivo de datagramas UDP puede producir la caída

del sistema. La víctima podría ser un Servidor de Nombres de Dominio (DNS) que es un sistema distribuido jerárquico cuya función es traducir las direcciones IP en etiquetas de servicio de red. DNS utiliza una base de datos distribuida y jerárquica que almacena información asociada a nombres de dominio en redes como Internet. Este protocolo está definido en RFC 1034 y en el RFC 1035 usa UDP como protocolo de capa transporte y trabaja en el puerto 53 por defecto (Saravanan & Asokan , 2012).

Otro tipo de ataque muy conocido es **SYN Flooding**. Es uno de los ataques DDoS que aparecieron por primera vez. Y hasta ahora es el más utilizado. La inundación SYN funciona aprovechando las debilidades del protocolo de control de transmisión (TCP). La figura 1 muestra el mecanismo del ataque de inundación SYN. El paquete SYN es un tipo de paquete en el Protocolo de Control de Transmisión (TCP) que requiere establecer una conexión entre dos hosts. Es una solicitud enviada por el host para hacer una conexión.

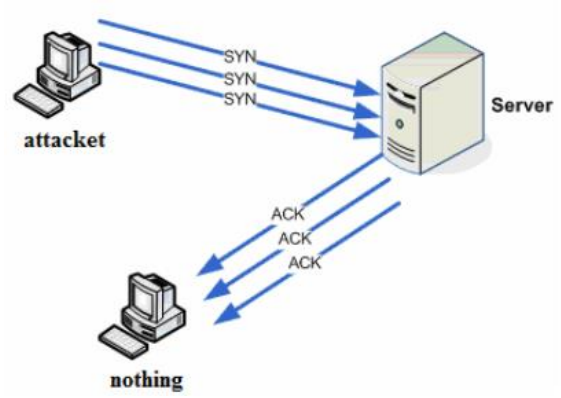


***Figura 2**. Mecanismo SYN Flooding (Sanmorino & Setiadi , 2013).*

2.3. Seguridad

La seguridad en la red se ha convertido en un esfuerzo sumamente importante y que presenta grandes desafíos para las organizaciones hoy en día. La finalidad es proteger la información confidencial e importante para brindar un servicio sin interrupciones evitando que diversas anomalías causen que los servicios se detengan (DK, Bhattacharyya, JK, & Kalita, 2013).

2.4. Mecanismos de Detección y Defensa

Los mecanismos de defensa y prevención de ataques arreglan los agujeros de seguridad en los hosts de Internet como protocolos débiles, planes de autenticación inadecuados, sistemas

informáticos no protegidos y sistemas operativos que se utilizan para originar un ataque DDoS (Gupta, Misra, & Joshi, 2012).

Los mecanismos de defensa identifican anomalías en la red conjuntamente ayudando a salvaguardar información relevante de los sistemas que se pretenden atacar. A pesar de la implementación de estos mecanismos, existen aún varios puntos indefensos que, con el desarrollo de nuevos ataques existentes en la actualidad, pueden ser atacados causando daños significativos. Los mecanismos de defensa también permiten analizar el tráfico de red y determinar si existe o no un ataque DDoS en algún lugar del sistema (Narváez, Romero, & Núñez, 2010).

Según (Javier Sanchez Gonzales, 2016), cuando un ataque DDoS no es masivo, una oportuna configuración del sistema operativo puede ser de utilidad para minimizar el ataque y habilitar nuevamente los servicios afectados. Una gran cantidad de servidores emplean el sistema operativo Linux, por lo tanto, se va a investigar sobre los parámetros del kernel de Linux que ayudan a minimizar los ataques maliciosos como un mecanismo de defensa. Entre ellos, se pretende analizar los siguientes:

- **Tcp_syncookies:** Protege de los ataques Syn_Flood; dependiendo del kernel, responde un segmento syn-ack creando una serie de números codificados que representa la IP origen y destino, el puerto y el timestamp de la petición recibida. Comando de inicio de las cookies: sysctl-w net.ipv4.tcp_syncookies=1 (Javier Sanchez Gonzales, 2016).
- **Ignore Broadcasts**: Protege de los ataques Smurf, envía paquetes ICMP a una dirección IP Broadcast de una dirección IP imitada inundando el servidor con el ataque Smurf; por lo tanto, para contrarrestar dicho ataque, se ejecuta el siguiente comando: sysctl-w net.ipv4.icmp_echo_ignore_broadcasts=1 (Javier Sanchez Gonzales, 2016).
- **Rp_filter**: Es la comprobación de los paquetes que acceden a una interfaz, basándose en una dirección original identificando así el ataque IP Spoofing: sysctl-w net.ipv4.conf.all.rp_filter=1 (Javier Sanchez Gonzales, 2016).

Una de las opciones para enfrentar los ataques DDoS es mediante la implementación de modelos de seguridad que utilizando diversas técnicas tanto para identificar las diversas anomalías que concurren en la red como para salvaguardar los servicios ofrecidos de un sistema. Esta da como resultado una disminución del tráfico de peticiones en el servidor.

Según (C. Rosales Garcia, 2011), la metodología que fue impartida para la detección de ataques maliciosos es denominada red bayesiana, que permite identificar incidencias, de los ataques en la red de datos, como principio importante menciona que los investigadores forenses optan por herramientas que ayudan a identificar la información para el análisis el cual tiene como resultado minimizar los ataques y realizar un análisis forense con el objetivo de conocer la forma de trabajo del atacante y sus técnicas. Se define los procesos a seguir en el diseño de la red bayesiana: Identificación de ataques DDoS, Diseño de red Bayesiana, Recolección de tráfico de red, Filtrado de tráfico de red, Pruebas de la metodología bayesiana, Ajustes al modelo de la red Bayesiana, Generación inferencial, Obtención de métricas finales.

Esta metodología fue diseñada principalmente para la obtención del tráfico de datos en la red, filtrando según el IDS Snort, permitiendo eliminar en tiempo real los ataques DDoS, siendo los ataques con mayor incidencia en Internet. A continuación, se presenta la funcionalidad de la red Bayesiana:

- Identificación del ataque DDoS.
- Emite un pronóstico al mismo.
- Envía alertas que determinan la incidencia detectada en la red.
- Emite una probabilidad de ocurrencia con un intervalo de confianza del 95 %.

Según (Sufian Hameed, 2016), menciona que el marco de detección de ataques DDoS en tiempo real obtenido por Hadoop (HADEC) posee cuatro fases importantes que son: Captura de trafico de red, generación de registros, transferencia de registros, detección de DDoS, notificación de resultados, captura de trafico de red y generación de registros.

En la detección de los ataques DDoS, HADEC proporciona una interfaz online que a través de la cual el administrador puede manipular el servidor con los parámetros según sea posible mantener una seguridad estable dentro de la entidad, el número de archivos a capturar antes

de iniciar la fase de detección y la ruta en la cual se va a guardar el archivo capturado, luego de que el administrador realice las configuraciones necesarias, el tráfico de Handdle inicia con la detección de tráfico de paquetes en tiempo real (Sufian Hameed, 2016).

HADEC utiliza un componente Tshark que permite realizar dicha captura, se ha desarrollado la utilidad de java based (Echo Class) que permite el desarrollo de una conexión con Tshark para que a través de ella pueda leer paquetes de salida de Tshark puestas en un archivo de registro. Una vez que el archivo sea desarrollado la clase Echo, alerta al administrador de tráficos de red que la detección se llevó a cabo exitosamente (Sufian Hameed, 2016).

Según el ajuste realizado en Tshark pudieron constatar la detección de los ataques, para lo cual emite lo más relevante de lo que se desarrolló en la fase de detección, incluyendo la información de: marcas de tiempo, IP de src, IP de dst, protocolo de paquete y un resumen sobre los encabezados de los paquetes que se listan a continuación son aquellos que se encuentran registrados en los archivos de registro: TCP (SYN), HTTP, UDP, ICMP.

Fase de transferencia de registro.

Luego de que se genera el archivo de registro, el controlador del tráfico emita sus alertas respectivas a la detección de ataques maliciosos.

El servidor de detección inicia un protocolo de copia de seguridad y envía desde el servidor a su sistema de archivo local funcionando principalmente como un NameNode denominada la parte central del clúster Hadoop y HDFS obteniendo la transferencia del registro almacenado localmente a HDFS exitosamente el servidor envía una alerta de recibo positivo (Sufian Hameed, 2016).

Según los dos tipos de metodologías analizadas se obtuvo conocimiento básico acerca de los mecanismos de defensa y herramientas que permiten analizar, notificar sobre los posibles ataques que han sido mencionados anteriormente.

2.5. ACL (Access Control List)

Según (Mifsud, 2015), ACLs un concepto de seguridad informática usado para fomentar la separación de privilegios. Es una forma de determinar los permisos de acceso apropiados a un determinado objeto, dependiendo de ciertos aspectos del proceso que hace el pedido.

Las ACLs permiten controlar el flujo del tráfico en equipos de redes, tales como routers y switches. Su principal objetivo es filtrar tráfico, permitiendo o denegando el tráfico de red de acuerdo con alguna condición. Sin embargo, también tienen usos adicionales, como, por ejemplo, distinguir "tráfico interesante" (tráfico suficientemente importante como para activar o mantener una conexión) en ISDN.

Objetivos de las ACLs

En resumen, los objetivos que se persiguen con la creación de ACL son:

- Limitar el tráfico de red y mejorar el rendimiento de la red. Al restringir el tráfico de vídeo, por ejemplo, las ACL pueden reducir ampliamente la carga de la red y en consecuencia mejorar el rendimiento de la misma.
- Controlar el flujo del tráfico. Las ACL pueden restringir el envío de las actualizaciones de enrutamiento. Si no se necesitan actualizaciones debido a las condiciones de la red, se preserva el ancho de banda.
- Proporcionar un nivel básico de seguridad para el acceso a la red. Las ACL pueden permitir que un host acceda a una parte de la red y evitar que otro acceda a la misma área. Por ejemplo, el host-1 se le permite el acceso a la red de producción, y al host-2 se le niega el acceso a esa red.
- Establecer qué tipo de tráfico se envía o se bloquea en las interfaces del router. Por ejemplo, permitir que se envíe el tráfico relativo al correo electrónico, y se bloquea el tráfico de ftp.
- Otorgar o denegar permiso a los usuarios para acceder a ciertos tipos de archivos, tales como FTP o HTTP.

Funcionamiento de las ACL

Para explicar el funcionamiento utilizaremos el software Cisco IOS.

El orden de las sentencias ACL es importante.

- Cuando el router está decidiendo si se envía o bloquea un paquete, el IOS prueba el paquete, verifica si cumple o no cada sentencia de condición, en el orden en que se crearon las sentencias.

- Una vez que se verifica que existe una coincidencia, no se siguen verificando otras sentencias de condición.

Por lo tanto, Cisco IOS verifica si los paquetes cumplen cada sentencia de condición de arriba hacia abajo, en orden. Cuando se encuentra una coincidencia, se ejecuta la acción de aceptar o rechazar y ya no se continúa comprobando otras ACL.

2.6. Syn Proxy

Syn Proxy es un proxy de paquetes TCP SYN. Puede usarse para proteger cualquier servidor TCP (como un servidor web) de inundaciones SYN y ataques DDoS similares. Syn Proxy es un módulo de filtro de red, en el kernel de Linux y Windows. Está optimizado para manejar millones de paquetes por segundo utilizando todas las CPU disponibles sin ningún bloqueo de concurrencia entre las conexiones. El efecto neto de esto es que los servidores reales no notarán ningún cambio durante el ataque. Las conexiones TCP válidas pasarán y se servirán, mientras que el ataque se detendrá en el firewall (Schabel, 2018).

¿Qué es un ACK?

Es un mensaje que el destino de la comunicación envía al origen de esta para confirmar la recepción de un mensaje. Si el mensaje está protegido por un código detector de errores y el dispositivo de destino posee además capacidad para procesar dicha información, el ACK también puede informar si se ha recibido de forma íntegra y sin cambios. De forma adicional, en protocolos de comunicaciones más complejos se definen diferentes ACK con información más compleja como peticiones de reenvió de ciertas tramas, información sobre incidencias en la red, etc (Schabel, 2018).

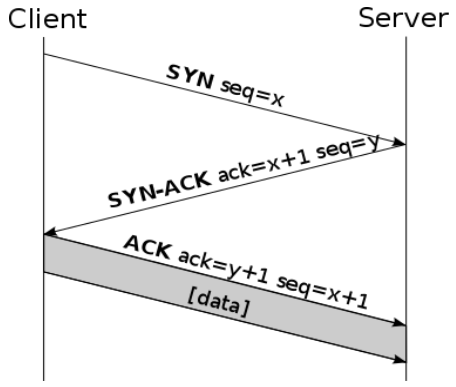


Figura 3. *Funcionalidad de ACK (Schabel, 2018).*

Secuencia de funcionalidad de Syn Proxy:

1. Cuando se utiliza una SYNPROXY, los clientes se conectan de forma transparente a la SYNPROXY. Por lo tanto, el protocolo de enlace TCP de 3 vías ocurre primero entre el cliente y SYNPROXY:
2. Los clientes envían TCP SYN al servidor A.
3. En el servidor de seguridad, cuando llega este paquete, se marca como DESCONECTADO.
4. El paquete TCP SYN sincronizado se entrega a SYNPROXY.
5. SYNPROXY obtiene esto y responde (como servidor A) con TCP SYN + ACK (UNTRACKED).
6. El cliente responde con TCP ACK (marcado como INVALID o UNTRACKED en iptables) que también se entrega a SYNPROXY.
7. Una vez que un cliente se ha conectado a SYNPROXY, SYNPROXY inicia automáticamente un protocolo de enlace TCP de 3 vías con el servidor real, falsificando el paquete SYN para que el servidor real vea que el cliente original está intentando conectarse:
8. SYNPROXY envía TCP SYN al servidor real A. Esta es una NUEVA conexión en iptables y ocurre en la cadena de SALIDA. La IP de origen del paquete es la IP del cliente.
9. El servidor real A responde con SYN + ACK al cliente.
10. SYNPROXY recibe esto y responde al servidor con ACK. La conexión ahora está marcada como ESTABLECIDA.
11. Una vez establecida la conexión, SYNPROXY deja el flujo de tráfico entre el cliente y el servidor.

Entonces, SYNPROXY puede usarse para cualquier tipo de tráfico TCP/UDP. Puede usarse tanto para el tráfico no cifrado como para el cifrado, ya que no interfiere con el contenido en sí.

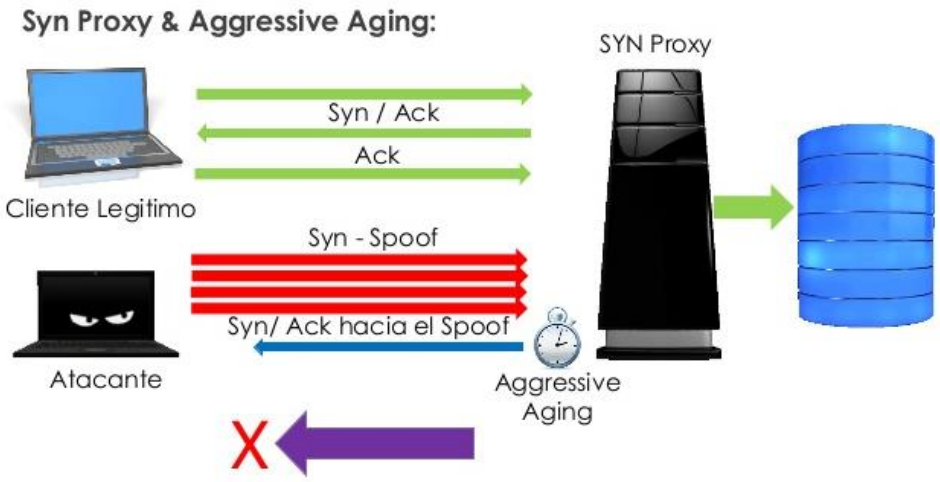


Figura 4. *Funcionalidad Syn Proxy (Pereira, 2018).*

2.7. Modelo

Según (Malena, 2013), un modelo es una representación formal de un sistema que se conecta a través de funciones matemáticas o lógicas de los parámetros del sistema en sí y sus relaciones con el mundo exterior. Estas funciones son de fundamental importancia porque describen las reglas de funcionamiento del sistema a ser simulado.

Un modelo es una representación de la realidad por medio de abstracciones. Los modelos enfocan ciertas partes importantes de un sistema (por lo menos, aquella que le interesan a un tipo de modelo específico), restándole importancia a otras.

La historia de la ciencia puede ayudarnos a comprender por qué el concepto de modelo no ha arraigado en la práctica científica de las ciencias sociales. El uso científico del término modelo se origina en el campo de las ciencias físicas y de las ciencias formales (lógica matemática) entre 1860 y 1900 (Armatte, 2016). En aquellos tiempos estaba en plena eclosión la disputa del método que instauró una controversia epistemológica profunda en torno a la cientificidad de las ciencias sociales. Tras las huellas del pensamiento de Dilthey y Rickert quedaron establecidas las categorías polares de la Methodenstreit que dividían a las ciencias naturales (Naturwissenschaften) y las ciencias del espíritu (Geisteswissenschaften) (Naishtat, 2014).

En estas coordenadas histórico-críticas cobra sentido el hecho que comenzar a hablar de modelo o, más precisamente, de modelos computacionales y modelos de simulación en

ciencias sociales implica necesariamente una apertura hacia un concepto y una forma de práctica científica que no se encuentra en el repertorio habitual de las disciplinas sociales y humanísticas. Más aún, la modelización y la simulación implican para el investigador social una apertura cultural hacia lenguajes y disciplinas que le son, en principio, ajenos (Roggero & Blanc, 2018).

El proceso de construcción de modelos

El análisis del sistema a través de un modelo implica que la representación del sistema que constituye el modelo ha de ser una representación manipulable. El ejercicio de construcción del modelo del sistema comienza por la construcción de un modelo conceptual del sistema, representación equivalente lógica aproximada del sistema real que, como tal, constituye una abstracción simplificada del mismo, que a continuación se traduce en un modelo apto para su ejecución en un ordenador. El proceso de modelización o construcción del modelo implica:

- Identificación de las entidades principales del sistema y de sus atributos característicos.
- Identificación y representación de las reglas que gobiernan el sistema que se quiere simular.
- Captación de la naturaleza de las interacciones lógicas del sistema que se modeliza.
- Verificación de que las reglas incorporadas al modelo son una representación válida de las del sistema que se modeliza.
- Representación del comportamiento aleatorio.

Una precaución importante a tener en cuenta cuando se construye un modelo es que ningún modelo es mejor que las hipótesis que encierra. Traducir el modelo a un modelo específico para ordenador consiste en representar el modelo conceptual mediante un lenguaje apto para su ejecución en un ordenador. Este proceso se simplifica cuando la representación se hace utilizando un lenguaje especializado orientado a problemas específicos. Las etapas del proceso de construcción del modelo se sintetizan en la Figura 5 (Garcia, 2010).

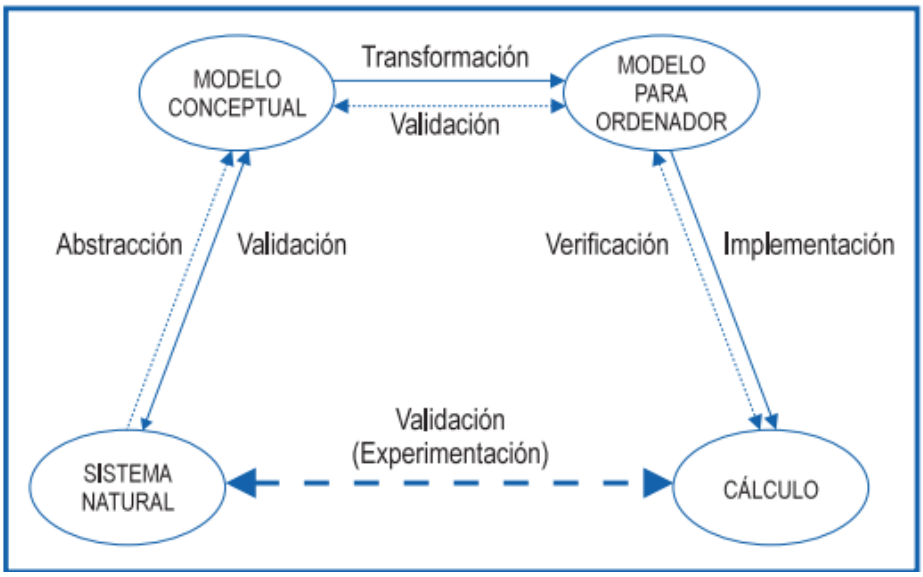


Figura 5. *Proceso de Modelización (Garcia, 2010).*

2.8. GNS3

GNS3 es un software utilizado por cientos de miles de ingenieros de redes a nivel mundial para emular, configurar, probar y solucionar problemas de redes virtuales y reales. GNS3 le permite ejecutar una pequeña topología que consta de solo unos pocos dispositivos en su computadora portátil, a aquellos que tienen muchos dispositivos alojados en múltiples servidores o incluso alojados en la nube (Telectrónika, 2018) .

En la actualidad GNS3 está activamente desarrollado y respaldado, cuenta con una comunidad en crecimiento de más de 800,000 miembros. GNS3 ha permitido a los ingenieros de red virtualizar dispositivos de hardware reales durante más de 10 años. Originalmente solo emulaba dispositivos Cisco que usaban software llamado Dynamips, GNS3 ahora ha evolucionado y admite muchos dispositivos de múltiples proveedores de red, incluidos conmutadores virtuales Cisco, Cisco ASA, Brocade v Routers, conmutadores Cumulus Linux, instancias Docker, HPE VSR, múltiples dispositivos Linux y muchos otros (Telectrónika, 2018).

Arquitectura de GNS3

La herramienta GNS3 consta de dos componentes de software:

- Software GNS3 todo en uno (GUI)
- Servidor/Máquina Virtual GNS3

Emulación y Simulación en GNS3

La herramienta GNS3 admite tanto dispositivos simulados como emulados.

- **Emulación en GNS3**
  - GNS3 imita o emula el hardware de un dispositivo y realiza ejecuciones de imágenes reales en el dispositivo virtual (Telectrónika, 2018).
- **Simulación en GNS3**
  - GNS3 simula las funciones y la funcionalidad de un dispositivo como un interruptor. No está ejecutando sistemas operativos reales, como Cisco IOS, sino más bien un dispositivo simulado desarrollado por GNS3 (Telectrónika, 2018).

Requerimientos del emulador, simulador de red GNS3

GNS3 es una plataforma que permite simular topologías de red con imágenes de IOs como Cisco y Juniper, entre otros. A continuación, se muestran los requerimientos para la instalación del software GNS3 (Telectrónika, 2018).

Compatibilidad con Windows

GNS3 es compatible con los siguientes sistemas operativos de Windows:

- Windows 7 SP1 (64 bit)
- Windows 8 (64 bit)
- Windows 10 (64 bit)
- Windows Server 2012 (64 bit)
- Windows Server 2016 (64 bit)

Requerimientos mínimos GNS3.

Tabla 2.1. *Requerimientos mínimos GNS.*

Ítem	Requerimientos Mínimos
Sistema Operativo	Windows 7 (64 bit) o superior
Procesador	2 o más núcleos lógicos
Virtualización	Se requieren extensiones de virtualización. Es posible que deba habilitar esto a través del BIOS de su computadora.
Memoria	4 GB RAM
Espacio en disco	1GB de espacio disponible (la instalación es < 200MB).
Notas adicionales	Es posible que necesite almacenamiento adicional para su sistema operativo e imágenes de los equipos.

Fuente: (Telectrónika, 2018).

Requerimientos óptimos GNS3.

Tabla 2.2. *Requerimientos óptimos GNS.*

Ítem	Requerimientos Óptimos
Sistema Operativo	Windows 7 (64 bit) o superior
Procesador	i7 CPU
Virtualización	8 o más núcleos lógicos – AMD-V / RVI Series o Intel VT-X / EPT
Memoria	Se requieren extensiones de virtualización. Es posible que deba habilitar esto a través del BIOS de su computadora.
Espacio en disco	32 GB RAM
Notas adicionales	Disco de Estado Sólido (SDD) 80 GB de espacio disponible
Notas adicionales	La virtualización de dispositivos consume mucho procesador y memoria, por lo tanto, mas es mejor, tener en cuenta si el dispositivo configurado correctamente supera la RAM y la potencia de procesamiento.

Fuente: (Telectrónika, 2018).

CAPÍTULO III

3. Metodología

La metodología utilizada en esta investigación es un estudio longitudinal, pues se obtiene datos en distintos momentos durante un período determinado, con la finalidad de examinar sus variaciones en el tiempo.

Los datos estadísticos se obtienen a partir de un escenario de simulación que implementa la topología de red de la UNACH y los ataques DDoS hacia la red. Esto permite analizar a detalle el comportamiento de este tipo de ataque, para poder detectarlo. Se manejarán dos escenarios de simulación. El primer escenario simulado no estará controlado por el mecanismo de detección del ataque de DDoS, y el segundo escenario simulado si estará controlado por el mecanismo de la detección del ataque de DDoS. La implementación de estos escenarios en un entorno de simulación ayudará a cuantificar el impacto en la red antes y un después del ataque DDoS.

Para la comprobación de la hipótesis se utilizará la prueba estadística T-Student para muestras relacionadas y un estudio longitudinal, siendo la variable fija la que crea dos medidas: una variable antes de aplicar el mecanismo de detección de ataques DDoS y una después de aplicar el mecanismo de detección de ataques DDoS en la red simulada de la UNACH. Luego se toma la variable aleatoria como variable de comparación numérica para el estudio.

3.1. Tipo De Estudio

3.1.1. Según el objeto de estudio.

- **Investigación Aplicada**

  Permite simular redes, así como también la aplicación de mecanismos de detección de ataques DDoS para poder obtener resultados de pre y post después de haber aplicado un ataque DDoS.

3.1.2. Según nivel de medición y análisis de la información.

- **Investigación Descriptiva**

Se realizó un análisis sobre las vulnerabilidades y riegos que conllevan los ataques DDoS y afectan la optimización de la red, así como también medir y evaluar el estado de la red simulada con la utilización de herramientas.

3.1.3. Según las Variables.

- **Investigación Cuasi - Experimental**

Se realizó el estudio de la red de la Universidad Nacional de Chimborazo en dos grupos, uno controlado y el otro experimental, simulando varios escenarios con la ayuda de diversas herramientas. El primer grupo no estará controlado por el mecanismo para la detección de ataques de DDoS, y el segundo si estará controlado por el mecanismo para la detección de ataques de DDoS. Esto permitirá comprobar el alcance que tiene este tipo de ataques que son reconocidos a nivel mundial por su efectividad y de esta manera poder comprobar la hipótesis de la investigación.

3.2. Población y Muestra

En la investigación no se dispone de población y muestra porque se va a trabajar con una red simulada que actualmente está en producción en la Universidad Nacional de Chimborazo**.**

3.3. Hipótesis

Ho: La aplicación del modelo de detección de ataques de DDoS **NO** permite mejorar la seguridad de las redes de la Universidad Nacional de Chimborazo.

Ha: La aplicación del modelo de detección de ataques de DDoS permite mejorar la seguridad de las redes de la Universidad Nacional de Chimborazo.

3.4. Operacionalización de Variables

Tabla 3.1. *Identificación de variables.*

VARIABLE	TIPO	DEFINICIÓN CONCEPTUAL	DIMENSIÓN	INDICADORES
La aplicación del modelo de detección de ataques de DDoS.	Independiente	Representación de la realidad por medio de abstracciones. Los modelos enfocan ciertas partes importantes de un sistema (por lo menos, aquella que le interesan a un tipo de modelo específico), restándole importancia a otras	• Conceptualización. • Experimentación.	- Validación. - Verificación.
El grado de mejora de seguridad en la red de la Universidad Nacional de Chimborazo.	Dependiente	Políticas y prácticas proyectadas para prevenir y supervisar el acceso no autorizado, el uso indebido, la modificación o la denegación de una red informática y sus recursos accesibles.	• Ataques Activos y pasivos. • Ataques por inundación de paquetes.	- Cantidad de paquetes procesados por día. - Nivel de vulnerabilidad por ataques DDoS. - Tiempos de respuesta del servidor.

3.5. Procedimientos

Para el estudio y la realización de un modelo para la detección de ataques DDoS se procedió a estudiar algunas técnicas y herramientas para mitigar estos ataques. Para ello se plantea el siguiente orden cronológico de acciones:

Primer paso:

Estudiar los ataques DDoS y mecanismos existentes de detección ante este tipo de ataques informáticos.

Segundo paso:

Diseñar la infraestructura de red de la UNACH, en un entorno de simulación para analizar el comportamiento del ataque DDoS y su mecanismo de detección.

Tercer paso:

Desarrollar el modelo de detección de ataques DDoS, en la red de la UNACH sobre un entorno de simulación.

Cuarto paso:

Analizar los resultados obtenidos sobre el entorno simulado, determinando en consecuencia el desempeño que posee el modelo de detección frente al ataque DDoS en la Red de la UNACH.

Quinto paso:

Desarrollar la guía para la aplicación del modelo de detección de ataques en un entorno real.

3.6. Escenario real de la infraestructura de red de la UNACH

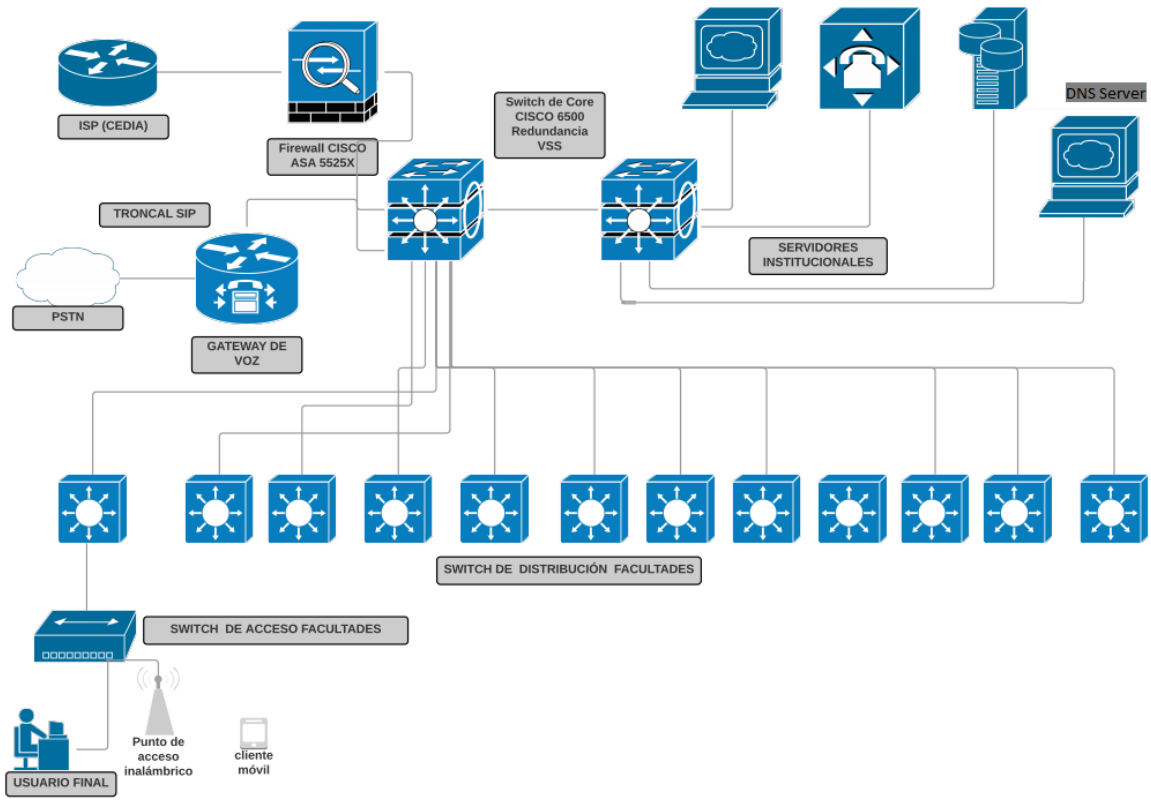


Figura 6. *Topología de red UNACH.*

La Topología de la Universidad Nacional de Chimborazo está compuesta por los dispositivos mostrados en la Fig. 6, considerando para el estudio y la simulación los dispositivos siguientes:

- Router:
  - Router ISP(CEDIA)
  - Router Gateway de voz: Troncal SIP, PSTN
- Firewall CISCO ASA 5525X
- Switch:
  - Switch de CORE CISCO 6500 redundancia VSS
  - Switchs de distribución de facultades
  - Switchs de acceso de facultades
- Servidores Institucionales
- Usuario Final:
  - Cliente: Host
  - Cliente: Móvil

CAPÍTULO IV

4. Resultados y Discusión

4.1 Análisis de Mecanismos de detección de ataques DDoS.

ACLs (Access Control List).

Las listas de control de acceso son usadas para fomentar la separación de privilegios, son una forma de determinar los permisos de acceso apropiados a un determinado objeto. Las ACLs permiten controlar el flujo del tráfico en equipos de redes, tales como routers y switches.

Funcionalidad.

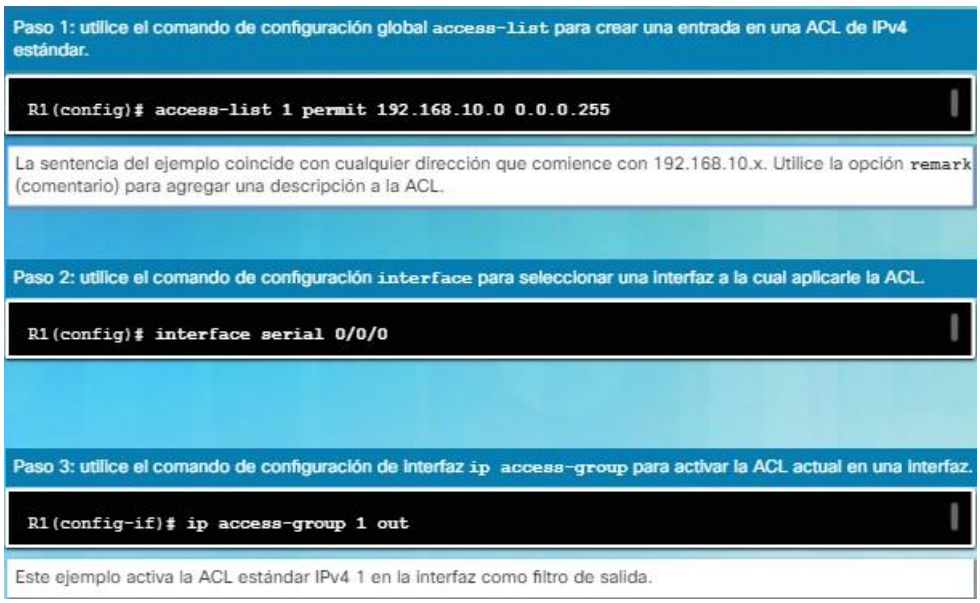


Figura 7. *Procedimiento para la configuración de ACL estándar.*

Tabla 4.1*. Cantidad de paquetes antes y después del mecanismo ACL.*

	Cantidad de paquetes antes	Cantidad de paquetes después
Primer día	9024	3502
Segundo día	8060	2050
Tercer día	7001	1820
Cuarto día	12010	4533
Quisto día	8124	3066
Promedio	8843,8	2994,2

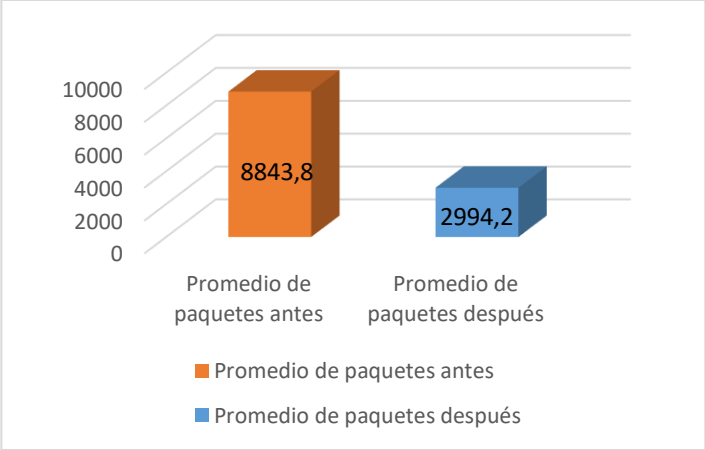


***Figura 8**. Promedio de paquetes antes y después del mecanismo ACL.*

Mediante el uso de la herramienta Wireshark para el análisis de paquetes se obtuvo un promedio de 8843,8 paquetes previo a la aplicación ACLs como mecanismo de detección, luego de ello se realizó el mismo análisis, pero una vez aplicado el mecanismo de detección de ataques, se obtuvo como resultado un promedio de 2994,2 y un 81,09% de paquetes menos como se puede apreciar en la fig. 8.

Syn Proxy.

Syn Proxy es un módulo de filtro de red, en el kernel de Linux y Windows, el cual puede usarse para proteger cualquier servidor TCP (como un servidor web) de inundaciones SYN y ataques DDoS similares.

***Tabla 4.2.** Cantidad e paquetes antes y después del mecanismo Syn proxy.*

	Cantidad de paquetes antes	Cantidad de paquetes después
Primer día	9024	2822
Segundo día	8060	1005
Tercer día	7001	911
Cuarto día	12010	2203
Quisto día	8124	1070
Promedio	8843,8	1602,2

De igual manera se hizo un análisis de paquetes con la ayuda de la herramienta Wireshark, se obtuvo un promedio de 8843,8 paquetes previo a la aplicación Syn Proxy como mecanismo de detección, luego de ello se realizó el mismo análisis, pero una vez aplicado el mecanismo de detección de ataques, se obtuvo como resultado un promedio de 1602,2 y un 88,4% de paquetes menos como se puede apreciar en la fig. 9.

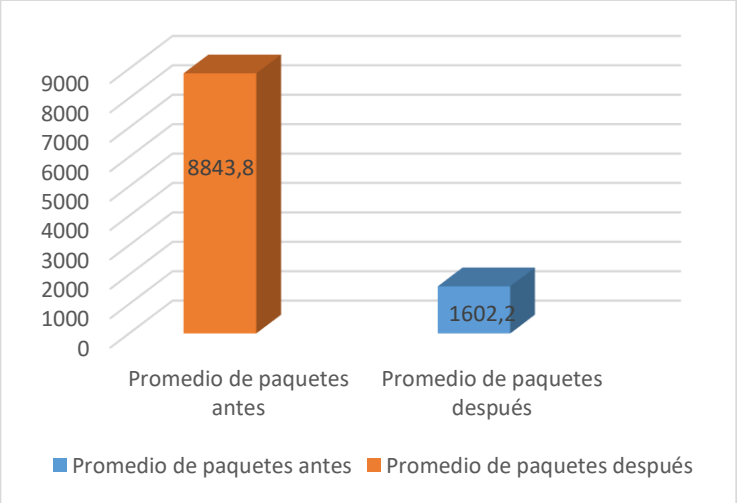


Figura 9. *Promedio de paquetes antes y después del mecanismo Syn Proxy.*

Una vez realizada la comparación entre los dos mecanismos **ACLS** y **Syn Proxy** bajo un ambiente de simulación (Fig. 8, Fig. 9) sobre los cuales se pudo determinar que el mejor mecanismo para la detección y mitigación de ataques DDoS es **Syn Proxy** con respecto al indicador que hace referencia a la cantidad de paquetes procesados.

4.2 Implementación de la infraestructura de red de la UNACH en un entorno de simulación

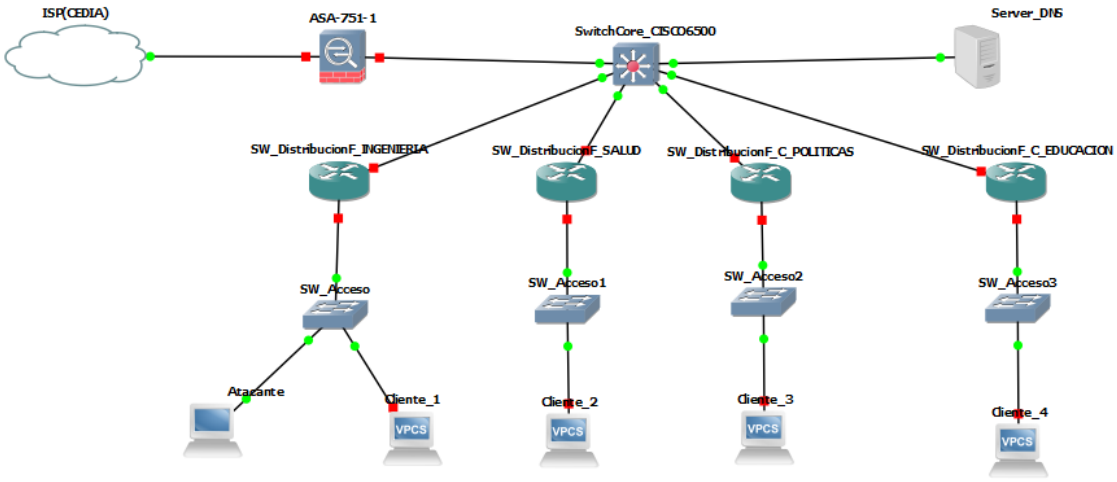


Figura 10. *Topología de Red Sobre el entorno de simulación GNS3.*

Este escenario se implementó sobre la herramienta GNS3 de acuerdo con la topología proporcionada por el administrador de red de la UNACH y de acuerdo con los dispositivos que se maneja.

4.3 Modelo para la detección de ataques DDoS

El modelo para la detección de ataques DDoS puede ser perfeccionado y modificado en el futuro dado que su estructura se debe ajustar a los constantes cambios que surgen de las organizaciones.

La particularidad del modelo que se presenta a continuación reside en su aspecto operativo y práctico, puesto que se considera su estructuración, formación e implementación bajo dos grandes fases.

- Fase de Elaboración
- Fase de Aplicación

Estas fases contemplan el conjunto de actividades que de ellas se desprenden y están ligadas mediante la secuencia de actividades que es necesario desarrollar a fin de elaborar y aplicar correctamente el modelo. La Fig. 12 presenta el modelo en forma esquemática. El esquema presentado resume y agrupa todas las actividades y se debe entender que muchas de ellas llevaran un ciclo continuo de mejora.

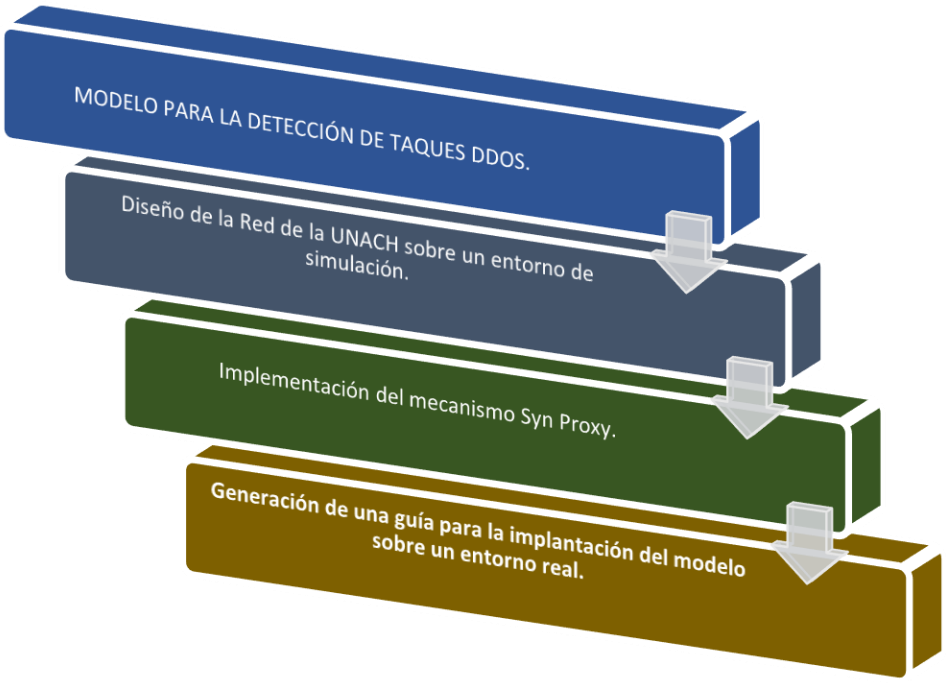


Figura 11. *Esquema del Modelo.*

A continuación, se presenta una descripción de cada una de las fases y actividades planteadas anteriormente:

1. **Diseño de la Red de la UNACH sobre un entorno de simulación.**

   El diseño de la red de la UNACH se lo realizo sobre el software de simulación de red GNS3 en el cual se usaron los siguientes dispositivos:

   - Un Switch de capa 3 para hacer referencia al Switch core que maneja la UNACH.

Figura 12*. Switch capa 3*

   - Firewall ASA 751-1 que interactúa con el ISP y el Switch core.

Figura 13. *Firewall ASA*

   - Un servido DNS el cual lo hemos alojamos sobre Windows server 2012 sobre una máquina virtual Virtual-Box.

Figura 14*. Servidor DNS.*

   - Router: SW_DistribucionF_INGENIERIA, SW_DistribucionF_SALUD, SW_DistribucionF_C_POLITICAS, SW_DistribucionF_C_POLITICAS

     En los switch de distribución se realizó todas las configuraciones de enrutamiento necesarias para conectar entre sí a toda la red.

Figura 15. *Switchs de distribución.*

- Host: clientes
- Host Atacante: El cual lo hemos alojado sobre una máquina virtual Virtual-Box con Kali Linux.

2. Implementación del mecanismo Syn Proxy.

Configurar un servidor proxy con Windows Server

2.1. Para comenzar, hacer clic en el menú de Windows en la sección "Control Panel".

Figura 16. *Panel de control Windows.*

2.2. Luego desde esta ventana dirigirse a "Redes e Internet" y dentro de esta sección, "Opciones de Internet".

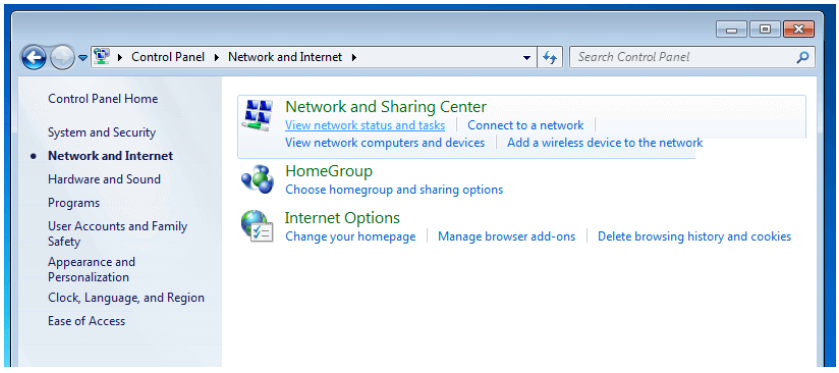


Figura 17. *Redes e Internet Windows.*

2.3. Se abre la ventana “Propiedades de Internet”.

Figura 18. *Propiedades de Internet Windows.*

2.4. En la pestaña “Conexiones” se encuentra el botón “LAN Settings”, que abre, la ventana “Ajustes de la red de área local”.

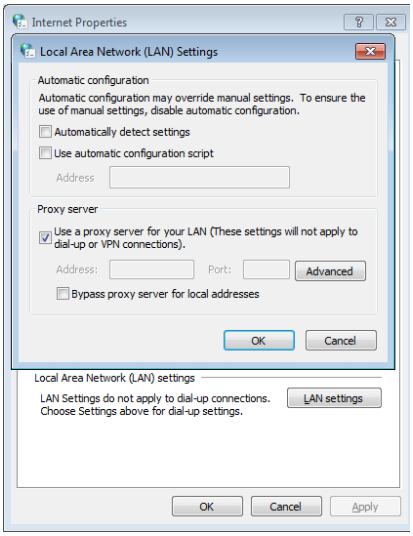


Figura 19. *Ajustes de la red de área local Windows.*

2.5. En este cuadro de diálogo se muestran tres posibilidades de configuración. Windows puede reconocer una configuración proxy de forma automática, utilizar un script de configuración automática o recurrir a la configuración manual.

2.6. Si el servidor proxy se tiene que configurar manualmente, es recomendable no marcar la casilla del reconocimiento automático de la configuración, pues esta

opción podría superponerse a la configuración manual. En su lugar y como se quiere configurar un servidor proxy se tiene que marcar la tercera opción, donde se introduce la dirección del proxy, así como el número del puerto.

2.7. Haciendo clic en "Avanzado" dirigirse al cuadro de diálogo "Ajustes proxy", donde se pueden configurar diversos servidores proxy para HTTP, HTTPS, FTP y SOCKS. Si todos los protocolos tienen que utilizar el mismo servidor solo hay que marcar la casilla "Use the same proxy server for all protocols". También es posible definir excepciones ("Exceptions"), de forma que se puede incluir en el campo de texto aquellas direcciones que no han de ser solicitadas con el servidor proxy.

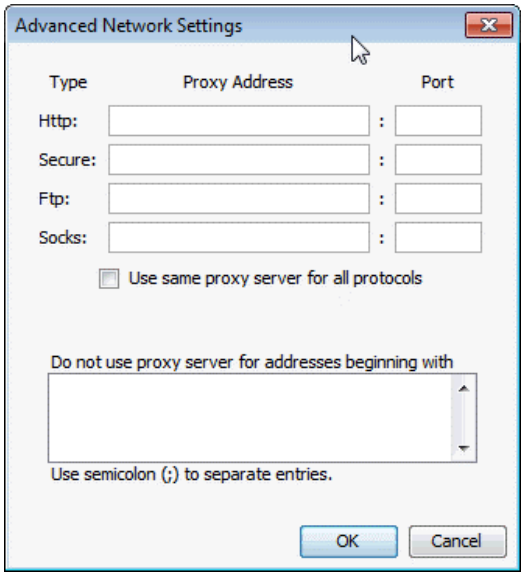


Figura 20. *Configuración Avanzada de red Windows.*

2.8. Los datos que necesitas para la configuración los recibes por parte del proveedor del servidor proxy.

2.9. Algunos servidores están protegidos con datos de acceso individuales. Su acceso requiere en este caso una cuenta de usuario personal con una contraseña.

4.4 Comprobación de hipótesis

La prueba de hipótesis es una regla que especifica si se puede aceptar o rechazar una hipótesis, con base en los datos de muestra, la prueba determina si se puede rechazar la hipótesis nula **Ho**. Para la verificación de la hipótesis se hará uso de la distribución T de Student con dos muestras relacionadas y porque las poblaciones son pequeñas. En el primer período, las observaciones y datos obtenidos servirán de control o testigo, para conocer los cambios que se susciten después de aplicar una variable experimental (Sarabia, 2014).

4.4.1 Planteamiento de hipótesis

Ho: La aplicación del modelo de detección de ataques de DDoS NO permite mejorar la seguridad de las redes de la Universidad Nacional de Chimborazo.

Ha: La aplicación del modelo de detección de ataques de DDoS permite mejorar la seguridad de las redes de la Universidad Nacional de Chimborazo.

4.4.2 Nivel de significación

El valor del nivel de significación va a ser de α=0.05 = 5%

4.4.3 Comprobación por indicador

4.4.3.1 Indicador: Cantidad de paquetes procesados por día.

Se aplicó un análisis de paquete de datos (Ataques DDoS) con la implementación del mecanismo de detención **Syn Proxy**, se realizó un estudio preliminar a la red de la UNACH y un estudio posterior a la aplicación del mecanismo de detección.

En la tabla 4.3 se plasma la cantidad de paquetes procesados por día antes y después de la implementación del mecanismo de detección de ataques Syn Proxy.

Tabla 4.3. *Cantidad de paquetes procesados por día.*

	Cantidad de paquetes antes	**Cantidad de paquetes después**
Primer día	9024	2822
Segundo día	8060	1005
Tercer día	7001	911
Cuarto día	12010	2203
Quinto día	8124	1070

Los datos del indicador (Cantidad de paquetes procesados por día) obtenidos en la red de la UNACH provienen de una **distribución normal**.

Una vez obtenido y verificado la normalidad de los datos obtenidos, en la tabla 4.4 se muestra la Prueba de T Student para muestras relacionadas, en la cual se obtiene un P-Valor de 0.008 y un nivel de significancia de 0.0082 con los cuales se aplicará criterios para aceptar o negar las Hipótesis nula o alternativa.

Tabla 4.4. *Prueba de Muestras emparejadas para el primer indicador*.

		Diferencias emparejadas							
				Media de error estándar	95% de intervalo de confianza de la diferencia				
		Media	Desviación estándar		Inferior	Superior	t	gl	Sig. (bilateral)
Par 1	Cantidad_paquetes_antes - Cantidad_paquetes_despues	1861,600	369,838	165,397	1402,385	2320,815	11,255	4	,00082

P-Valor = 0.008	<=	**α=0.05**

4.4.3.2 Indicador: Nivel de vulnerabilidad por ataques DDoS.

Se realizó ataques DDoS en un periodo de 5 días en horas determinadas en la red de la UNACH, por consiguiente, se ha realizado una escala para medir el nivel de vulnerabilidad que es de 1 a 4 bajo, 5 a 7 medio y de 8 a 10 alto el nivel de vulnerabilidad. De igual manera se realizó un estudio preliminar y un estudio posterior a la aplicación del mecanismo de detección de ataques que ayuda a mitigar este tipo de vulnerabilidades. (Tabla 4.6)

Tabla 4.5. *Cantidad de ataques antes*

	Numero ataques 8 AM	**Numero ataques 12 PM**	**Numero ataques 6 PM**	**Cantidad ataques por día**
Día 1	3	5	4	12
Día 2	2	3	2	7
Día 3	5	1	3	9
Día 4	1	2	5	8
Día 5	4	4	2	10

Tabla 4.6. *Cantidad ataques después.*

	Número ataques 8 AM	Número ataques 12 PM	Número ataques 6 PM	Cantidad ataques por día
Día 1	1	2	2	5
Día 2	1	1	1	3
Día 3	3	0	1	4
Día 4	1	1	2	4
Día 5	2	1	0	3

Tabla 4.7. *Nivel de vulnerabilidad antes y después.*

	Nivel de vulnerabilidad Antes	Nivel de Vulnerabilidad Después
Día 1	12	5
Día 2	7	3
Día 3	9	4
Día 4	8	4
Día 5	10	3

Los datos del indicador (Nivel de vulnerabilidad por ataques DDoS) provienen de una **distribución normal.**

En la tabla 4.8 se plasma la Prueba de T Student para muestras relacionadas, en la cual se obtiene un P-Valor de 0.01 y un nivel de significancia de 0.001 con los cuales se aplicará criterios para aceptar o negar las Hipótesis nula o alternativa.

Tabla 4.8. *Prueba de Muestras emparejadas para el segundo indicador.*

Prueba de muestras emparejadas								
		Diferencias emparejadas						
	Media	Desviación estándar	Media de error estándar	95% de intervalo de confianza de la diferencia		t	gl	Sig. (bilateral)
				Inferior	Superior			
Par 1	Nivel_vulnerabilidad_Antes - Nivel_vulnerabilidad_Después	5,400	1,517	,678	3,517	7,283	7,962	4 ,001

P-Valor = 0.01	<=	**α=0.05**

4.4.3.3 Indicador: Tiempos de respuesta del servidor.

Se realizó un análisis de tiempos de respuesta hacia el servidor DNS con cinco herramientas de software distintas (Tabla 4.9) y se ha realizado una escala para medir el nivel de vulnerabilidad que es de 1 a 4 bajo, 5 a 7 medio y de 8 a 10 alto, de igual manera se realizó un estudio preliminar al servidos DNS y un estudio posterior a la aplicación del mecanismo de detección Syn Proxy. (Tabla 4.10)

Tabla 4.9. *Análisis de herramientas para medir tiempos de respuestas.*

HERRAMIENTA	DESCRIPCIÓN	PLATAFORMA
DNS Benchmarck	Es capaz de probar el rendimiento, fiabilidad, seguridad y velocidad de los servidores DNS de su base de datos y plasmarlo a través de informes gráficos y estadísticas.	Windows XP, Windows Vista, Windows 7, y para Linux y Mac OS X
Pingdom Tool	Pingdom ofrece tiempo de actividad rentable y confiable y monitoreo de rendimiento para su sitio web.	Multiplataforma
Google Analytics	Google Analytics es una herramienta de analítica web de la empresa Google. Ofrece información agrupada del tráfico que llega a los sitios web según la audiencia, la adquisición, el comportamiento y las conversiones que se llevan a cabo en el sitio web.	Multiplataforma
Namebench	Herramienta que nos ayuda a encontrar cuales son los mejores servidores DNS, tanto en rapidez como en seguridad.	Windows, Mac y Linux
DNSPerf.com	Herramienta de Nominum que permiten realizar pruebas de carga a servidores DNS e interpretar sus datos con informes y gráficas.	Unix, GNU/Linux, Windows

Tabla 4.10. *Tiempos de respuesta del servidor.*

Nombre Herramienta	Tiempos de respuesta antes	Tiempos de respuesta después
DNS Benchmarck	8,5 s	3,2 s
Pingdom Tool	8,4 s	4,5 s
Google Analytics	8,5 s	3,6 s
Namebench	9,0 s	5,1 s
DNSPerf.com	8,9 s	3,3 s

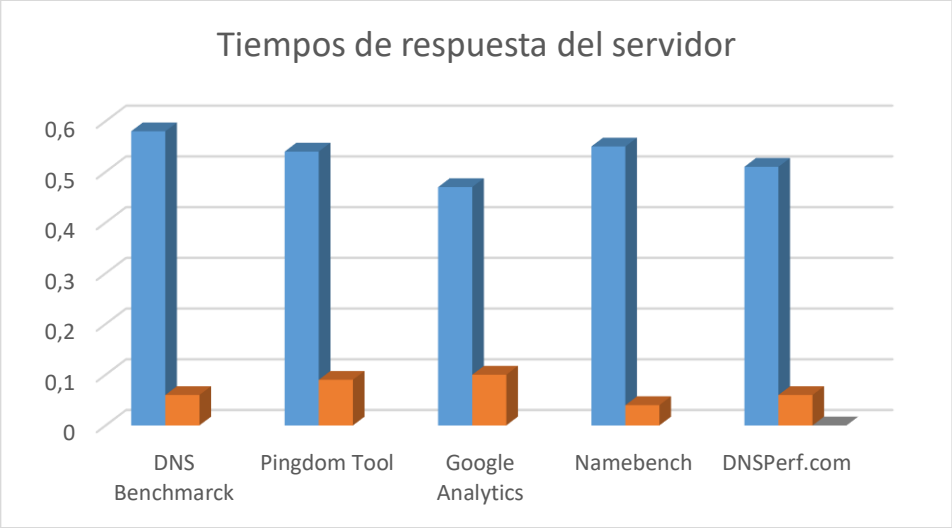


Figura 21. *Tiempos de respuesta del servidor.*

Los datos del indicador (Tiempos de respuesta del servidor) provienen de una **distribución normal**.

En la tabla 4.11 se muestra la Prueba de T Student para muestras relacionadas, en la cual se obtiene un P-Valor de 0.013 y un nivel de significancia de 0.0013 con los cuales se aplicará criterios para aceptar o negar las Hipótesis nula o alternativa.

Tabla 4.11. *Prueba de Muestras emparejadas para el tercer indicador.*

Prueba de muestras emparejadas									
		Diferencias emparejadas							
					95% de intervalo de confianza de la diferencia				
		Media	Desviación estándar	Media de error estándar	Inferior	Superior	t	gl	Sig. (bilateral)
Par 1	Tiempos_respuesta_antes - Tiempos_respuesta_despues	4,7200	,7887	,3527	3,7407	5,6993	4,278	4	,0013

P-Valor = 0.013	**<=**	**α=0.05**

4.4.4 Análisis e Interpretación de indicadores.

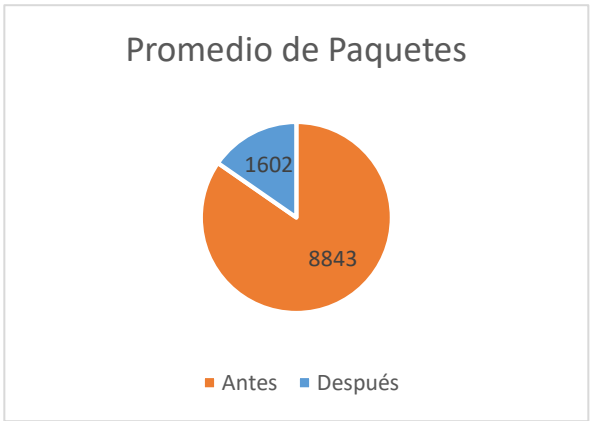


Figura 22. *Promedio de paquetes.*

- Mediante el análisis de paquetes utilizando la herramienta wireshark se obtuvo un promedio de 8843 paquetes que se trasmitían por la red de la UNACH previamente a la aplicación del mecanismo de detención de ataques DDoS. Se realizó el mismo estudio dicha herramienta, pero una vez aplicado el mecanismo de detención de ataques DDoS, se obtuvo como resultado un promedio de 1602 paquetes que se trasmitían, dando como resultado favorable que se disminuyó en un 81.09% esta vulnerabilidad de la red y con una significancia de 0,00082. Cabe recalcar que no se reduce en su totalidad pues este tipo de ataques trabajan en modo pasivo, el mecanismo de detección de ataques no es efectivo en su totalidad. Por lo cual, se establece que la **aplicación del modelo de detección de ataques de DDoS SI permite mejorar la seguridad de las redes de la Universidad Nacional de Chimborazo**.

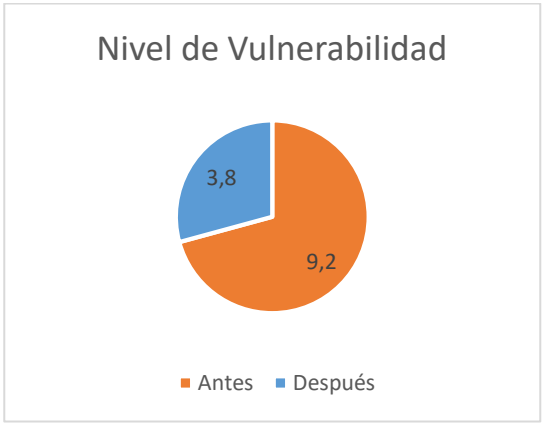
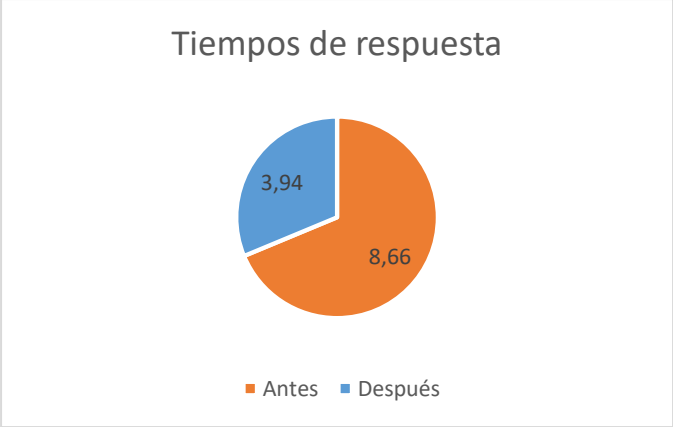


Figura 23. *Nivel de vulnerabilidad.*

- El estudio se realizó en un periodo de cinco días en determinadas horas para medir el nivel de vulnerabilidad de la red de la UNACH con la cual se obtuvo un promedio de nivel de vulnerabilidad de 9.2 sobre 10 previo a la aplicación del mecanismo de detección. Una vez aplicado el mecanismo de detección de ataques en la red se realizó los mismos ataques y como resultado se obtuvo un promedio de 3.8 sobre 10, teniendo como conclusión que se disminuyó el nivel de vulnerabilidad en un 58.7% y con una significancia de 0.001. Por lo cual, se establece que **la aplicación del modelo de detección de ataques de DDoS SI permite mejorar la seguridad de las redes de la Universidad Nacional de Chimborazo.**



- El estudio se hizo en base a las cinco herramientas para medir los tiempos de respuesta del servidor DNS con lo cual se obtuvo un promedio de tiempo de respuesta de 8.66 sobre 10 previo a la aplicación del mecanismo de detección. Una vez aplicado el mecanismo de detección de ataques en la red se realizó los mismos ataques y como resultado se obtuvo un promedio de 3.94 sobre 10, teniendo como conclusión que se disminuyó el tiempo de respuesta del servidor DNS en un 55% y con una significancia de 0.013. Por lo cual, se establece **que la aplicación del modelo de detección de ataques de DDoS sobre un entorno simulado SI permite mejorar la seguridad de las redes de la Universidad Nacional de Chimborazo.**

CAPÍTULO V

5. Conclusiones y recomendaciones

5.1 Conclusiones

- Luego de realizar el análisis de mecanismos para la detección de ataques DDoS y su aplicación en los diferentes escenarios, se determina que el mejor mecanismo es Syn Proxy, ya que es un mecanismo (Appliance) que se coloca antes del servidor real y efectivamente este niega el acceso a las conexiones nuevas e ilegítimas.
- La implementación del entorno simulado mediante el uso de GNS3 permitió utilizar versiones reales de IOS de los equipos de la infraestructura de Red de la UNACH garantizando el análisis de la aplicación del modelo y el comportamiento de los dispositivos al momento de simular los ataques DDoS y aplicar los mecanismos de detección establecidos.
- Una vez aplicado los cálculos estadísticos de los datos obtenidos en cada uno de los indicadores propuestos y después de la aplicación del mecanismo para la detección de ataques DDoS sobre el entorno de simulación, se determina que en el primer indicador Cantidad de paquetes procesados por día disminuye en un 81.09%; el segundo indicador que es Nivel de vulnerabilidad por ataques DDoS disminuye en un 58.7%; el tercer indicador denominado Tiempos de respuesta del servidor mejora los tiempos de respuesta del servidos DNS en un 55%, permitiendo decidir que el mecanismo para la detección de ataques aplicado si permite mejorar la seguridad del servidor DNS simulado en la red de la UNACH.
- De acuerdo con la auditoría técnica realizada y el modelo para la detección de ataques DDoS desarrollado en base a las vulnerabilidades detectadas en la red de la Universidad Nacional de Chimborazo sobre un entorno de simulación, se concluye que se podrá mejorar la seguridad en al menos un 65%.
- La guía desarrollada para la aplicación del modelo para la detección de ataques DDoS sobre un entorno real pasa a ser una fuente valiosa de información sobre como mitigar este tipo de ataques, la cual podrá ser utilizada por el administrador de red de la UNACH para su futura implementación.

5.2 Recomendaciones

- Tomar este trabajo de investigación como guía para las instituciones de nivel superior, para contar con una alternativa de solución ante ataques de DDoS y a su vez generar planes de contingencia en caso de otro tipo de ataques informáticos.
- Para poder realizar pruebas de seguridad informática en los equipos de una red se recomienda el software Kali Linux en los Host que actúan como atacante, pues es una herramienta que permite realizar auditorías informáticas, posee un abanico de herramientas todas ellas destinadas a realizar pruebas, diagnósticos y comprobaciones de aspectos importantes para evaluar la seguridad informática de los equipos destinados.
- Para obtener resultados más precisos donde mejore la calidad del mecanismo de detección de ataques DDoS, se puede seguir con la mejora del modelo desarrollado y estableciendo distintos indicadores que se adapten según al tipo de red que se vaya a analizar y a su vez incluir nuevos métodos de detección de ataques de DDoS.
- Toda institución educativa o entidad pública que maneje una red datos corporativa debe constar con una guía para la implementación de modelos y políticas de seguridad mediante los cuales se permita establecer normativas para los usuarios que hagan uso de los servicios que brinda la institución.

6. Bibliografía

Armatte, M. (Enero de 2016). *researchgate*. Obtenido de https://www.researchgate.net/publication/277268687_La_Nocion_de_Modelo_en_las_Ciencias_Sociales

Biazus, & Branco.M. (12 de Diciembre de 2016). Subvertendo um sistema de detecção de intrusão: Caso prático utilizando Snort e Nmap. *RIC-Revista de Informação Contábil.*, 38-58. Obtenido de https://repositorio.ufsc.br/handle/123456789/171434

C. Rosales Garcia, M. A. (Mayo de 2011). Identificación de ataques de DDoS en redes de datos a través de un modelo basado en una red bayesiana. *IPN, Repositorio Digital-Mediateca*, 1-77. Obtenido de http://repositoriodigital.ipn.mx/handle/123456789/12652

Cooke, E., Jahanian, F., & McPherson, D. (2014). The Zombie Roundup: Understanding, Detecting, and Disrupting Botnets. *Arbor Networks*. Obtenido de https://www.usenix.org/legacy/event/sruti05/tech/full_papers/cooke/cooke_html/

DK, Bhattacharyya, JK, & Kalita. (2013). A Machine Learning Perspective. *CRC Press: Taylor y Francis Group, Boca Raton*. Obtenido de http://103.4.94.107:8080/xmlui/bitstream/handle/123456789/17621/Network_Anomaly_Detection_A_Machine_Learning_Perspective.pdf?sequence=1

Feinstein.L, Schnackenberg.D, Balupari.R, & Kindred.D. (22-24 de Abril de 2013). Statistical Approaches to DDoS Attack Detection and Response. *Proceedings of the DARPA Information Survivability Conference and Exposition*. Obtenido de http://ieeexplore.ieee.org/abstract/document/1194894/?reload=true

Garcia, M. (2010). Discrete Event Simulation Methodologies. *Proceedings of the Winter Simulation Conference*.

Gupta, B., Misra, M., & Joshi, R. (2012). *www.tandfonline.com*. Obtenido de https://www.tandfonline.com/doi/full/10.1080/21642583.2017.1331768

Hoque.N, D. B. (22 de Febrero de 2016). FFSC a novel measure for low rate and high rate DDoS attack detection using multivariate data analysis. doi:10.1002/sec.1460

Hoque.N, Dhruba.K, Bhattacharyya, & Jugal.K. (2015). Botnet in DDoS Attacks_ Trends and Challenges. *IEEECOMMUNICATIONSURVEYS & TUTORIALS, 17*, 2242–2270. Obtenido de http://www.cs.uccs.edu/~jkalita/papers/2015/HoqueNazrulEEETutorials&Surveys2015.pdf

J.Arzamendia, & F.Lopez. (19 de Noviembre de 2016). Obtenido de https://www.researchgate.net/publication/310496769

J.Mirkovic, & Reiher, P. (2014). A taxonomy of DDoS attack and DDoS defense mechanisms. *ACM SIGCOMM Computer Communication Review, 34*, 39-53. Obtenido de https://www.eecis.udel.edu/~sunshine/publications/ccr.pdf

Javier Sanchez Gonzales, B. M. (Junio de 2016). Ciberseguridad mecanismos de ataque y defensa mas extendidos. *El Instituto Nacional de Tecnologías de la Comunicación (INTECO)*. Recuperado el febrero de 2011, de https://www.incibe.es/extfrontinteco/img/File/intecocert/EstudiosInformes/cert_inf_seguridad_analisis_trafico_wireshark.pdf

Juliette Dromard, V. B. (24 de Febrero de 2017). Experimental evaluation of algorithms for online network characterizations ONTIC: D4.3. *HAL-LAAS Open Archive Laboratory for Analysis and Architecture of Systems*. Obtenido de https://hal.laas.fr/hal-01476103

Lau, F., Stuart, R., & Michael , S. (06 de Agosto de 2012). *Distributed Denial of Service Attacks*. Obtenido de https://ieeexplore.ieee.org/abstract/document/886455/authors#authors

Liu, X., Yang, X., & Lu, Y. (2018). To filter or to authorize: Network-layer DoS defense against. *Proceedings of the ACM SIGCOMM 2008 Conference on*, 195–206.

Malena, G. (2013). Approcci Basati su tecniche di simulazione ad eventi discreti.

Mifsud, E. (30 de Septiembre de 2015). *http://recursostic.educacion.es*. Obtenido de http://recursostic.educacion.es/observatorio/web/es/component/content/article/1065-listas-de-control-de-acceso-acl?start=3

Molina Lorena, F. A. (Junio de 2015). Ataques Distribuidos de Denegación de Servicios: modelación y simulación con eventos discretos. *Conference: XV Jornada Internacional de Seguridad Informática – ACIS*, 277. doi:10.13140/RG.2.1.5123.5687

Naishtat. (2014). *POLIST*. Obtenido de https://journals.openedition.org/polis/10568?lang=pt#ftn2

Narváez, D., Romero, C., & Núñez, M. (2010). Evaluación de ataques de Denegación de servicio DoS y DDoS, y mecanismos de protección. *Revista DECC Report, Tendencias en Computación.* Obtenido de http://journal.espe.edu.ec/index.php/geeks/article/view/249/226

Pereira, D. (2018). Mitigacion de ataques DDoS. *SecPro*, 42.

Roggero, P., & Blanc, S. (2018). *http://www.scielo.org.mx*. Obtenido de http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0185-19182015000300227

Saman Zargar, J, J., & D, T. (2013). Hadoop Based Defense Solution to Handle Distributed Denial of Service (DDoS) Attacks. *Scientific Research*.

Sanmorino, A., & Setiadi , Y. (2013). DDoS Attack Detection Method and Mitigation Using. *ResearchGate*, 5.

Sarabia, J. M. (2014). *Distribuciones multivariantes con distribuciones condicionadas t de Student.* España.

Saravanan , K., & Asokan , R. (Enero de 2012). *International Journal of Computer Science, Engineering and Information Technology*. Obtenido de https://www.researchgate.net/publication/51988470_Distributed_Denial_of_Service_DDOS_Attacks_Detection_Mechanism

Schabel, L. (14 de Abril de 2018). *GitHub*. Obtenido de https://github.com/firehol/firehol/wiki/Working-with-SYNPROXY

Sufian Hameed, U. A. (Julio de 2016). Efficacy of Live DDoS Detection with Hadoop. *Network Operations and Management*. doi:10.1109/NOMS.2016.7502848

Telectrónika. (29 de Abil de 2018). *TelecTronika.com*. Obtenido de https://telectronika.com/articulos/que-es-gns3/

7. ANEXOS

Anexo N.º 1: Escenarios de la metodología de la investigación.

Escenario 1: Ataques DDoS en la red de la Universidad Nacional de Chimborazo.

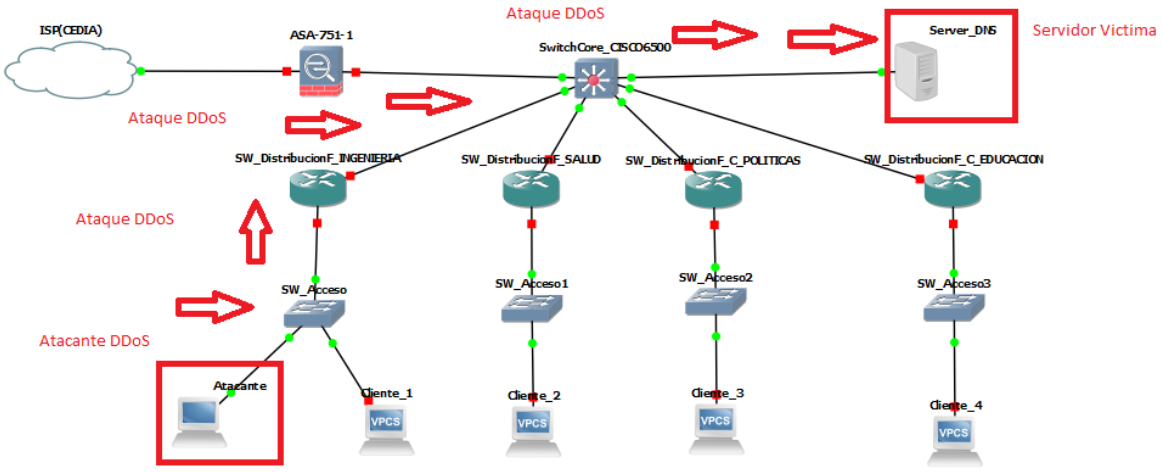


Figura 24. *Topología de Ataques DDoS en la red de la UNACH.*

Con la utilización de un ordenador que estará conectado como un cliente más de la red el cual actúa como atacante, se realizó un ataque DDoS el cual se trasmite por la red de la UNACH, luego se realizó un análisis de paquetes para lo cual se utilizó la herramienta WireShark, con el objetivo de conocer la vulnerabilidad del servidos DNS ante este tipo de ataques.

Escenario 2: Implementación de ACLs como mecanismo de detección.

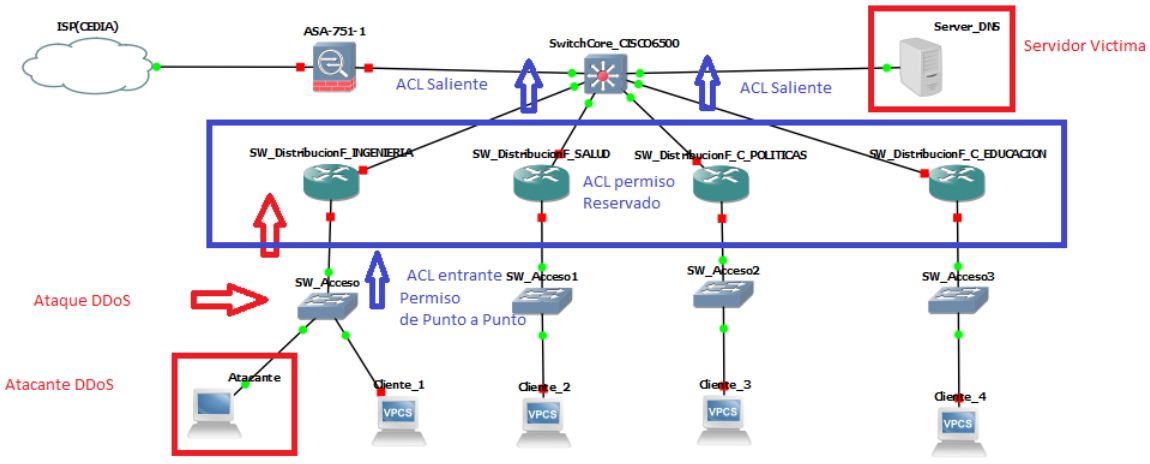


Figura 25. *Implementación de ACLs como mecanismo de detección.*

De igual manera se utilizó de un ordenador que estará conectado como un cliente más de la red el cual actúa como atacante, adicionalmente se implementó ACLs (Access Control List) como mecanismo de detección de ataques DDoS, para lo cual se realizaron configuraciones avanzadas de ACLs robustas, con el objetivo de medir la eficiencia de dicho mecanismo sobre la red de la UNACH.

Escenario 3: Implementación de Syn Proxy como mecanismo de detección.

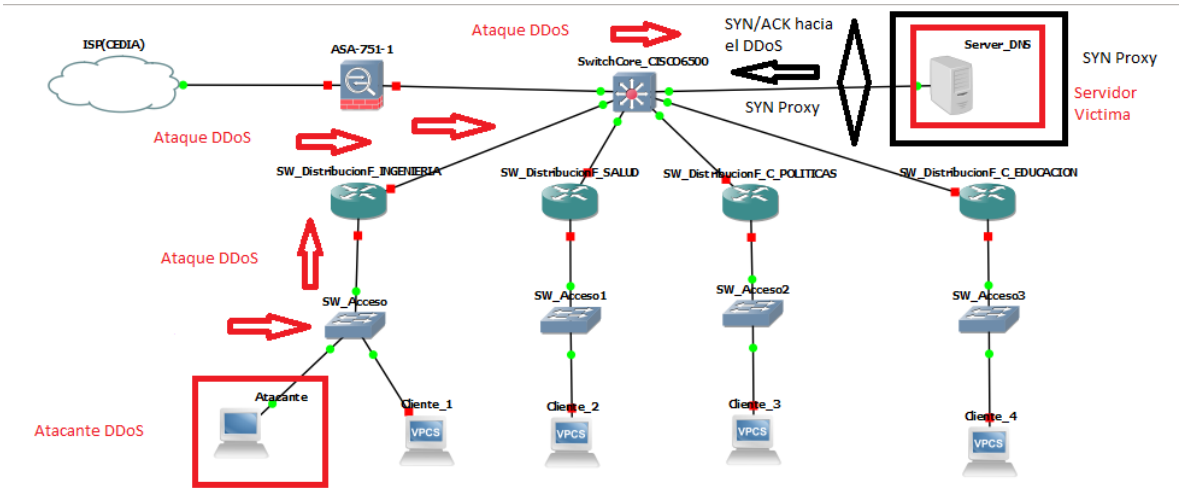


Figura 26. *Implementación de Syn Proxy como mecanismo de detección.*

Finalmente también se utilizó de un ordenador que estará conectado como un cliente más de la red el cual actúa como atacante, adicionalmente se implementó Syn Proxy como mecanismo de detección de ataques DDoS, para lo cual este mecanismo se lo puede configurar en Linux o a su vez en Windows como es nuestro caso ya que el servidor DNS se encuentra alejado en Windows server 2012, se realizaron las configuraciones necesarias para implementar Syn Proxy, con el objetivo de medir la eficiencia de dicho mecanismo sobre la red de la UNACH.

Anexo N.º 2: Resultados de Metodología Research.

Tabla 7.1. *Resultados de la Metodología Research.*

Query	Google scholar	IEEE	Microsoft Academic Search	Scopus	Total
Mecanismos de defensa	56	18	1	6	81
Attacks DDoS	120	11	7	5	143
Seguridad Informática	2	0	1	2	5
Mitigación de ataques	2	245	2	1	250

Tabla 7.2. *Criterios exclusión del método Research.*

APLICANDO CRITERIOS DE EXCLUSION	**GOOGLE ACADÉMICO**
Ataques DDoS SERVIDOR "ataque ddos" -CLIENTE	11
taxonomía de un ataque DDoS "ataques ddos" -CLIENTE	1
mitigación ataque DDoS "ataques DDoS" -cliente	9
mecanismo de defensa frente los ataques DDoS en DNS DDoS "ataques DDoS" -cliente	3
tipos de ataques ddos en dns ataques ataques OR DDoS OR ataques OR DDoS "tipos de ataques" -cliente -cliente	21
defensa en dns para ataques ddos ataques OR DDoS "mecanismos de defensa" -cliente	8
comportamiento del dns ante ataques DDoS DNS "ataques DDoS" -cliente	4
TOTAL	**57**
APLICANDO CRITERIOS DE EXCLUSION	**IEEE**
DDoS Journals & Magazines Early Access Articles Show: open access Year: 2017	2
DDoS in DNS Show: all Year :2016-2017	11
TOTAL	13
APLICANDO CRITERIOS DE EXCLUSION	**OTROS**
DDoS Attacks	3
Mitigación de ataques DDoS -clientes	1
Implementación de ataques DDoS -clientes	2
Mecanismos de defensa frente a DDoS -cliente	5
TOTAL	**11**

Anexo N.º 3: Comprobación de la hipótesis

A continuación, se muestra tablas y figuras en las cuales se detalla las pruebas de normalidad para cada indicador.

Resultados de paquetes procesados por día

Tabla 7.3. *Resumen de procesamiento de casos.*

	Casos					
	Válido		Perdidos		Total	
	N	Porcentaje	N	Porcentaje	N	Porcentaje
Cantidad_paquetes_antes	5	100,0%	0	0,0%	5	100,0%
Cantidad_paquetes_despues	5	100,0%	0	0,0%	5	100,0%

Fuente: *Herramienta IBM SPSS Statistics versión 24.0.0.0*

Tabla 7.4. *Descriptivos.*

			Estadístico	Error estándar
Cantidad_paquetes_antes	Media		8843,80	853,981
	95% de intervalo de confianza para la media	Límite inferior	6472,77	
		Límite superior	11214,83	
	Media recortada al 5%		8770,28	
	Mediana		8124,00	
	Varianza		3646415,200	
	Desviación estándar		1909,559	
	Mínimo		7001	
	Máximo		12010	
	Rango		5009	
	Rango intercuartil		2987	
	Asimetría		1,474	,913
	Curtosis		2,593	2,000

Cantidad_paquetes_despues	Media		1602,20	385,131
	95% de intervalo de confianza para la media	Límite inferior	532,91	
		Límite superior	2671,49	
	Media recortada al 5%		1572,83	
	Mediana		1070,00	
	Varianza		741628,700	
	Desviación estándar		861,179	
	Mínimo		911	
	Máximo		2822	
	Rango		1911	
	Rango intercuartil		1555	
	Asimetría		,873	,913
	Curtosis		-1,682	2,000

Fuente: *Herramienta IBM SPSS Statistics versión 24.0.0.0*

Tabla 7.5. *Pruebas de normalidad primer indicador.*

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Cantidad_paquetes_antes	,262	5	,200*	,870	5	,266
Cantidad_paquetes_despues	,332	5	,076	,820	5	,116

*. Esto es un límite inferior de la significación verdadera.
a. Corrección de significación de Lilliefors

Cantidad_paquetes_antes
Cantidad_paquetes_antes Gráfico de tallo y hojas
Frecuencia Stem & Hoja
1,00 7. 0
2,00 8. 01
1,00 9. 0
1,00 Extremos (>=12010)

Ancho del tallo: 1000
Cada hoja: 1 caso(s)

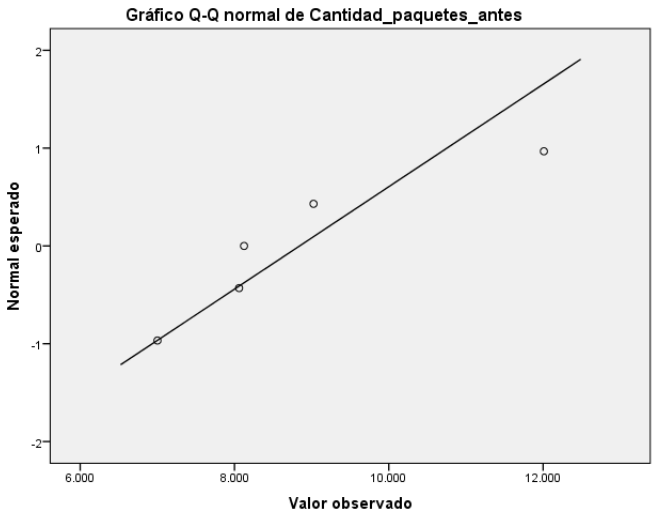


Figura 27. *Gráfico Q-Q normal cantidad de paquetes antes.*

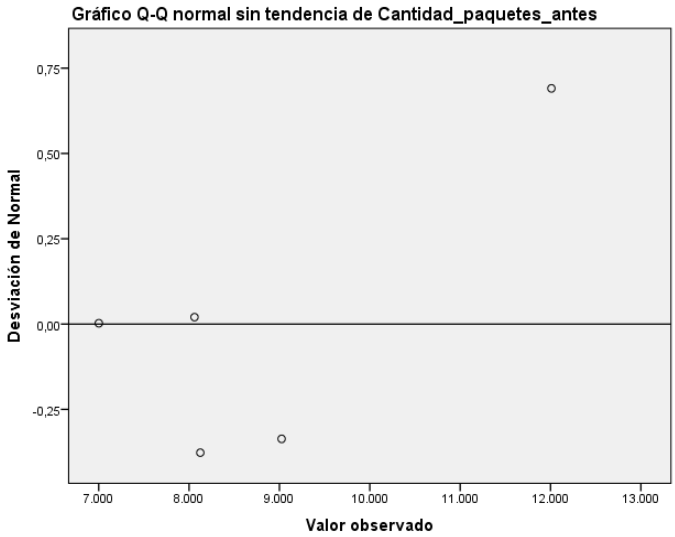


Figura 28. *Q-Q normal sin tendencias cantidad de paquetes antes.*

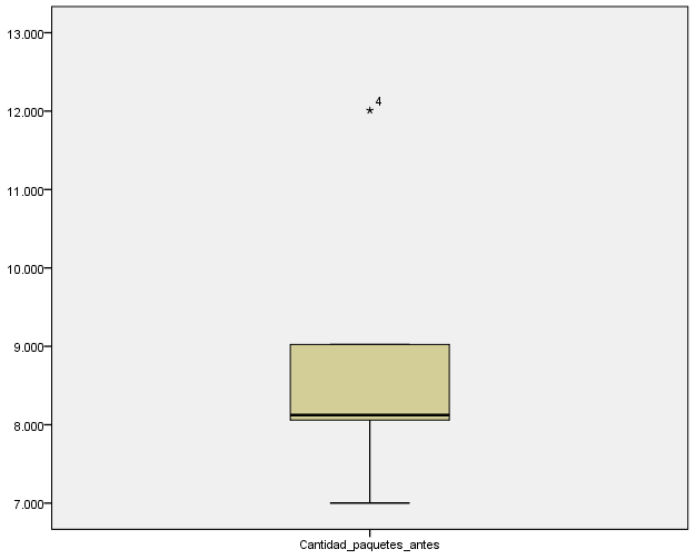


Figura 29. *Cantidad de paquetes antes.*

Cantidad de paquetes después
Cantidad_paquetes_despues Gráfico de tallo y hojas
Frecuencia Stem & Hoja

1,00	0. 9
2,00	1. 00
2,00	2. 28

Ancho del tallo: 1000
Cada hoja: 1 caso(s)

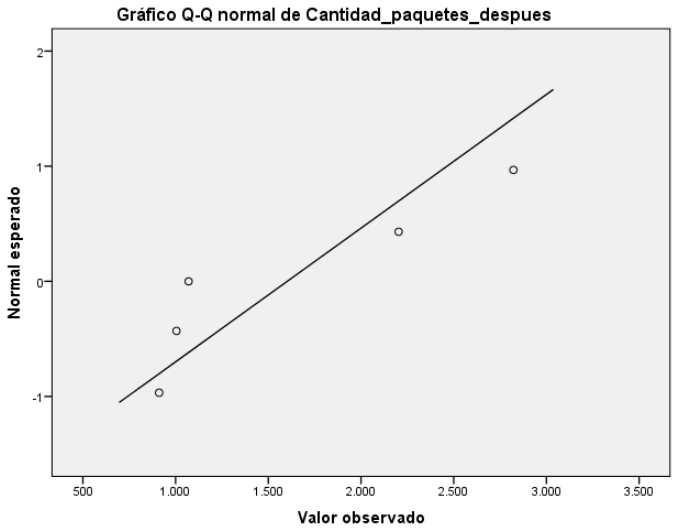


Figura 30. *Gráfico Q-Q normal cantidad de paquetes después.*

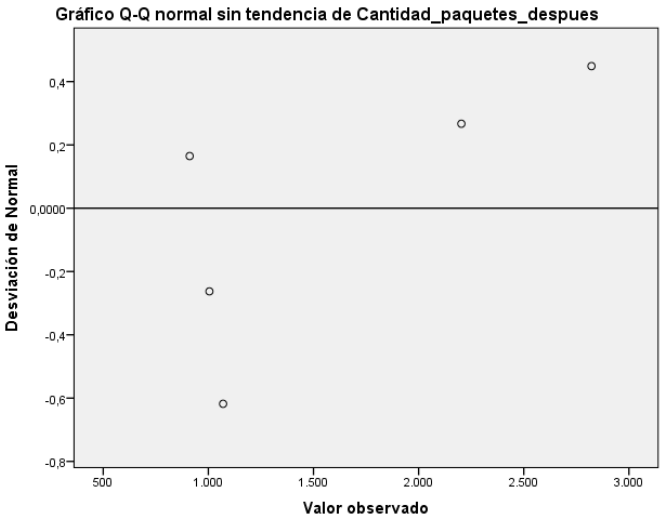


Figura 31. *Q-Q normal sin tendencias cantidad de paquetes después.*

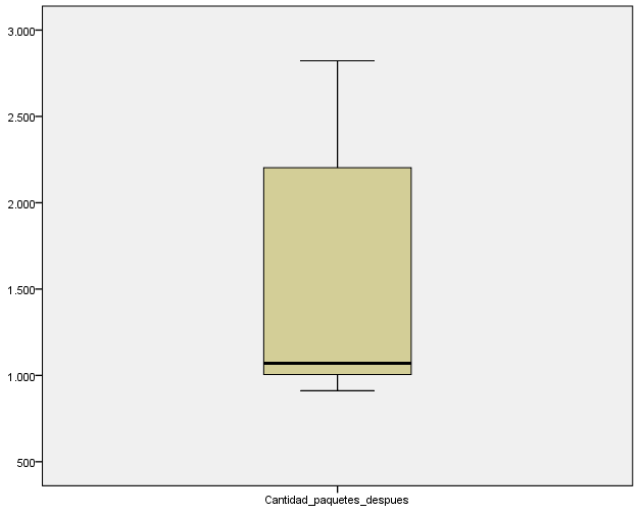


Figura 32. *Cantidad de paquetes después.*

Resultados de Nivel de vulnerabilidad por ataques DDoS

Tabla 7.6. *Resumen de procesamiento de casos.*

	Casos					
	Válido		Perdidos		Total	
	N	Porcentaje	N	Porcentaje	N	Porcentaje
Nivel_vulnerabilidad_Antes	5	100,0%	0	0,0%	5	100,0%
Nivel_vulnerabilidad_Después	5	100,0%	0	0,0%	5	100,0%

Fuente: *Herramienta IBM SPSS Statistics versión 24.0.0.0*

Tabla 7.7. *Descriptivos.*

			Estadístico	Error estándar
Nivel_vulnerabilidad_Antes	Media		9,20	,860
	95% de intervalo de confianza para la media	Límite inferior	6,81	
		Límite superior	11,59	
	Media recortada al 5%		9,17	
	Mediana		9,00	
	Varianza		3,700	
	Desviación estándar		1,924	
	Mínimo		7	
	Máximo		12	
	Rango		5	
	Rango intercuartil		4	
	Asimetría		,590	,913
	Curtosis		-,022	2,000
Nivel_vulnerabilidad_Después	Media		3,80	,374
	95% de intervalo de confianza para la media	Límite inferior	2,76	
		Límite superior	4,84	
	Media recortada al 5%		3,78	
	Mediana		4,00	
	Varianza		,700	
	Desviación estándar		,837	
	Mínimo		3	
	Máximo		5	
	Rango		2	
	Rango intercuartil		2	
	Asimetría		,512	,913
	Curtosis		-,612	2,000

Fuente: *Herramienta IBM SPSS Statistics versión 24.0.0.0*

Tabla 7.8. *Pruebas de normalidad segundo indicador.*

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Nivel_vulnerabilidad_Antes	,141	5	,200*	,979	5	,928
Nivel_vulnerabilidad_Después	,231	5	,200*	,881	5	,314

*. Esto es un límite inferior de la significación verdadera.

a. Corrección de significación de Lilliefors

Nivel_vulnerabilidad_Antes

Nivel_vulnerabilidad_Antes Gráfico de tallo y hojas

Frecuencia	Stem & Hoja
3,00	0. 789
2,00	1. 02

Ancho del tallo: 10
Cada hoja: 1 caso(s)

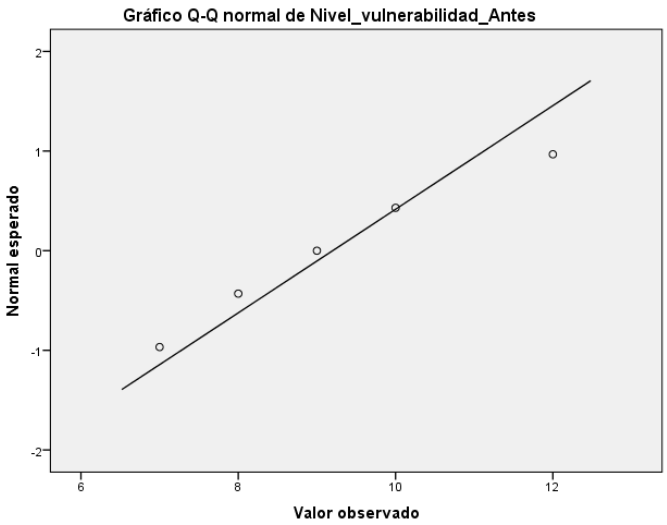


Figura 33*. Gráfico Q-Q normal nivel de vulnerabilidad antes.*

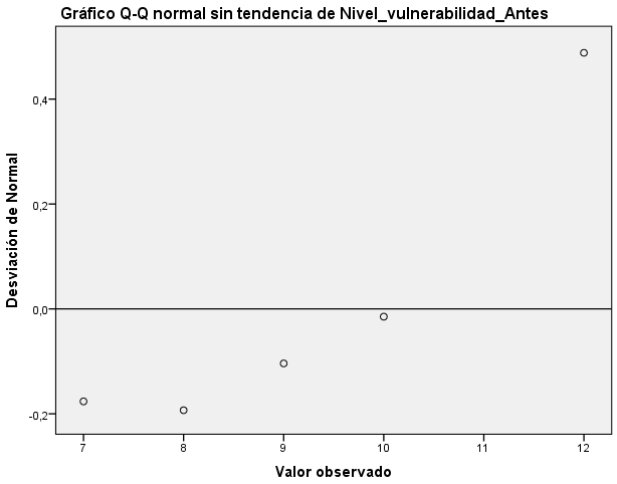


Figura 34*. Q-Q normal sin tendencias nivel de vulnerabilidad antes.*

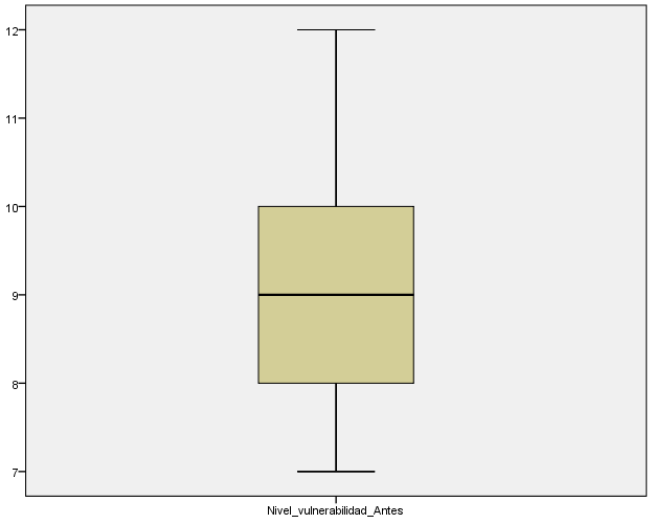


Figura 35*. Nivel de vulnerabilidad antes*

Nivel_vulnerabilidad_Después

Nivel_vulnerabilidad_Después Gráfico de tallo y hojas

Frecuencia Stem & Hoja

2,00	3. 00
2,00	4. 00
1,00	5. 0

Ancho del tallo: 1
Cada hoja: 1 caso(s)

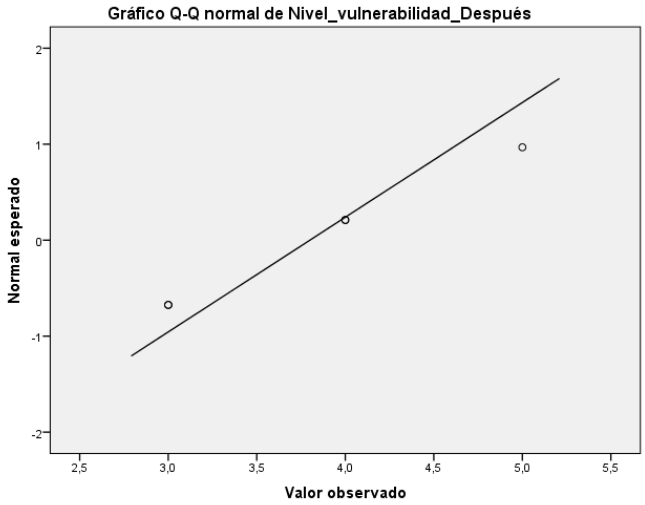


Figura 36. *Gráfico Q-Q normal nivel de vulnerabilidad después.*

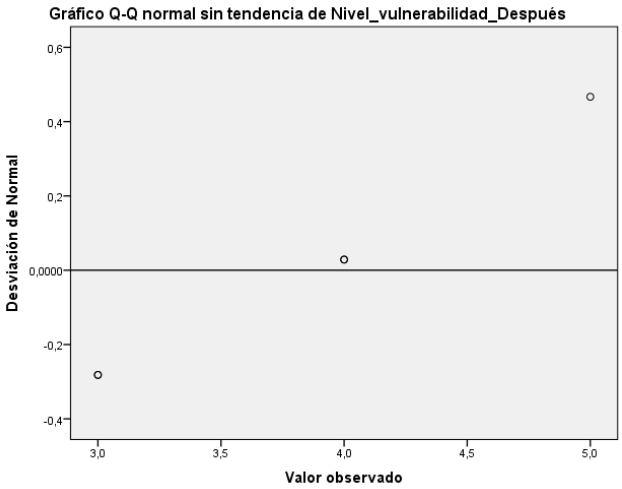


Figura 37. *Q-Q normal sin tendencias nivel de vulnerabilidad después.*

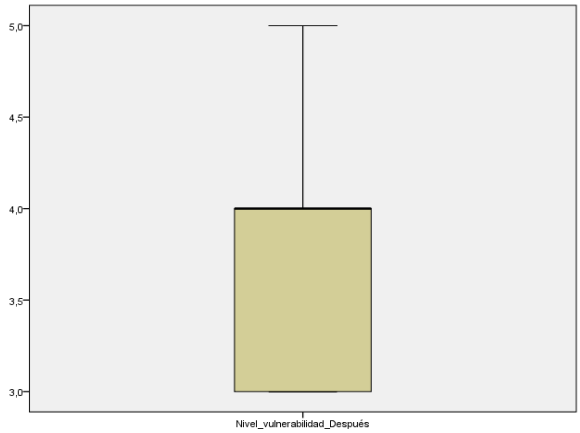


Figura 38. *Nivel de vulnerabilidad después.*

Resultados Tiempos de respuesta del servidor

Tabla 7.9. *Resumen de procesamiento de casos.*

	Casos					
	Válido		Perdidos		Total	
	N	Porcentaje	N	Porcentaje	N	Porcentaje
Tiempos_respuesta_antes	5	100,0%	0	0,0%	5	100,0%
Tiempos_respuesta_despues	5	100,0%	0	0,0%	5	100,0%

Fuente: *Herramienta IBM SPSS Statistics versión 24.0.0.0*

Tabla 7.10. *Descriptivos.*

			Estadístico	Error estándar
Tiempos_respuesta_antes	Media		8,660	,1208
	95% de intervalo de confianza para la media	Límite inferior	8,325	
		Límite superior	8,995	
	Media recortada al 5%		8,656	
	Mediana		8,500	
	Varianza		,073	
	Desviación estándar		,2702	
	Mínimo		8,4	
	Máximo		9,0	
	Rango		,6	
	Rango intercuartil		,5	
	Asimetría		,578	,913
	Curtosis		-2,708	2,000
Tiempos_respuesta_despues	Media		3,940	,3696
	95% de intervalo de confianza para la media	Límite inferior	2,914	
		Límite superior	4,966	
	Media recortada al 5%		3,917	
	Mediana		3,600	
	Varianza		,683	
	Desviación estándar		,8264	
	Mínimo		3,2	
	Máximo		5,1	
	Rango		1,9	
	Rango intercuartil		1,5	
	Asimetría		,760	,913
	Curtosis		-1,596	2,000

Fuente: *Herramienta IBM SPSS Statistics versión 24.0.0.0*

Tabla 7.11. *Pruebas de normalidad tercer indicador.*

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Tiempos_respuesta_antes	,323	5	,096	,840	5	,166
Tiempos_respuesta_despues	,260	5	,200*	,882	5	,319

*. Esto es un límite inferior de la significación verdadera.

Tiempos_respuesta_antes

Tiempos_respuesta_antes Gráfico de tallo y hojas

Frecuencia	Stem & Hoja
1,00	8. 4
3,00	8. 559
1,00	9. 0

Ancho del tallo: 1,0

Cada hoja: 1 caso(s)

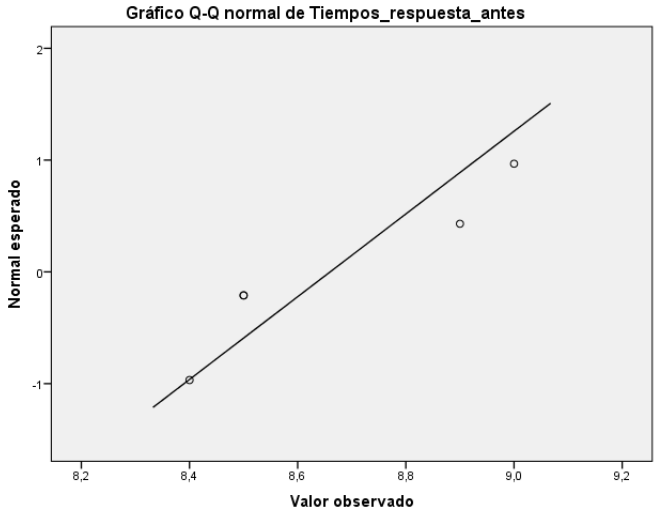


Figura 39. *Gráfico Q-Q normal de tiempos de respuesta antes.*

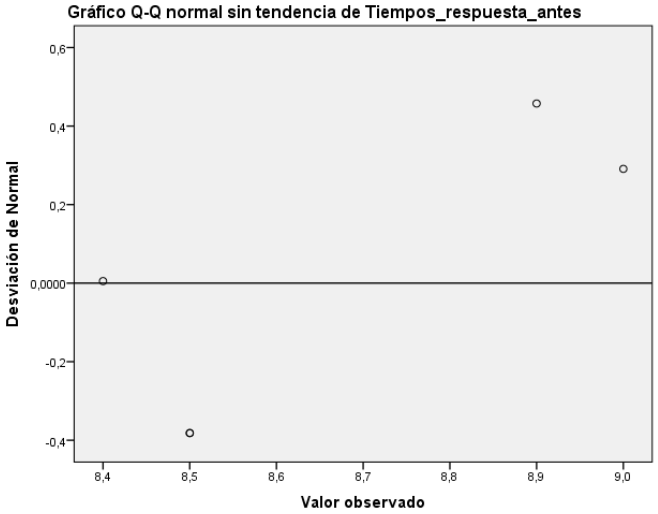


Figura 40. *Gráfico Q-Q normal sin tendencia de tiempos de respuesta antes.*

Figura 41. *Tiempos de respuesta antes.*

Tiempos_respuesta_después

Tiempos_respuesta_despues Gráfico de tallo y hojas

Frecuencia Stem & Hoja

3,00 3. 236

1,00 4. 5

1,00 5. 1

Ancho del tallo: 1,0

Cada hoja: 1 caso(s)

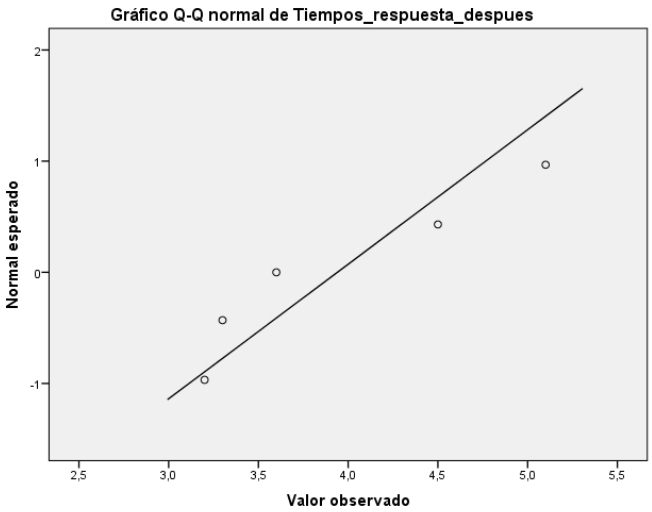


Figura 42. *Gráfico Q-Q normal de tiempos de respuesta después.*

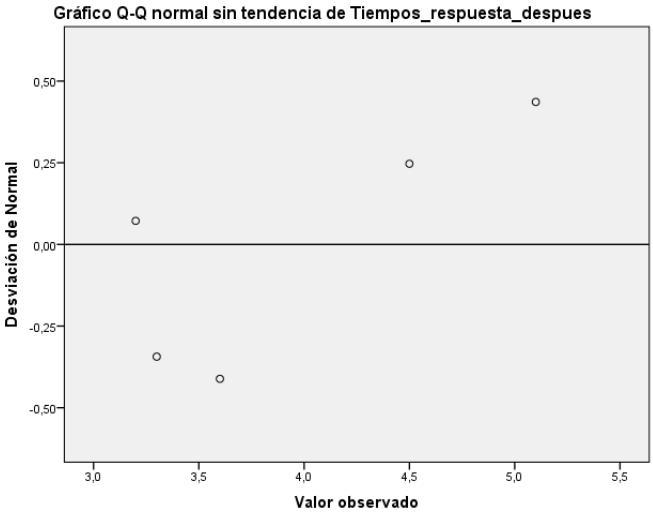


Figura 43. *Gráfico Q-Q normal sin tendencia de tiempos de respuesta después.*

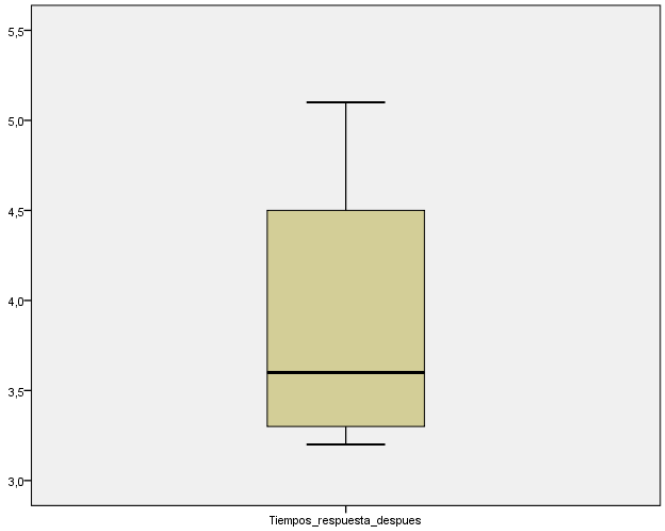


Figura 44. *Tiempos de respuesta después.*

Anexo N.º 4: Guía para la implementación del modelo para la detección y mitigación de ataques DDoS en un entorno real.

Universidad Nacional de Chimborazo

Riobamba, febrero de 2019

Control del Documento

Datos del documento	
Titulo:	**Guía para la implementación del modelo para la detección y mitigación de ataques DDoS en un entorno real.**
Versión:	V 1
Cliente:	Universidad Nacional de Chimborazo

Control del Cambios

Versión	Fecha	Autores	Descripción
1.1	15/02/2019	Marlon Marcelo Miranda Martínez	
1.1	15/02/2019	Dany Xavier Bonifaz Herrera	

1. Objetivo

Implementar el modelo de detección de ataques de DDoS en un entorno real.

2. Metodología para el desarrollo de la guía

Para la elaboración de la guía, deben seguirse los siguientes pasos:

- Como detectar ataques DDoS mediante el análisis de Trafico.
- Como implementar el mecanismo Syn Proxy para la detección y mitigación de ataques DDoS.
- Como medir el comportamiento del tráfico de nuestra red una vez implementado el mecanismo Syn Proxy.

3. Desarrollo de la guía.

1. Como detectar ataques DDoS mediante el análisis de Tráfico.

Para la detección de ataques, realizar un análisis de tráfico haciendo uso de la herramienta WIRESHARK. Wireshark es un analizador de protocolos open-source diseñado por Gerald Combs y que actualmente está disponible para plataformas Windows y Unix.

1.1.-La captura de datos lo realizaremos haciendo uso de un HUB.

Haciendo uso del HUB conectar en el mismo segmento de red donde se encuentra el servidor DNS, donde al tratarse de un medio de comunicación compartido, todo el tráfico entre el switch y el servidor podrá ser analizado desde nuestro equipo.

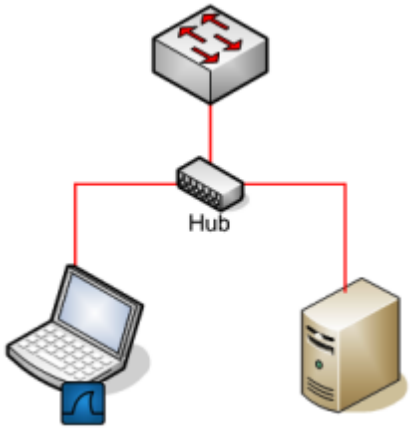


Figura 45. *Modo de captura.*

Para analizar el tráfico de la red, aremos uso de la herramienta WIRESHARK ya que esta no permitirá apuntar a los paquetes que realmente sean sospechosos, y a su vez nos permitirá hacer un análisis de la traza del intercambio de paquetes en general, para tener una mayor visibilidad acerca del ataque sufrido.

1.2.- Captura de tráfico con WIRESHARK.

Una vez configurado nuestro HUB, lanzar Wireshark como root/administrador. Para iniciar la captura seleccionamos la interfaz en el menú Capture >> Interfaces.

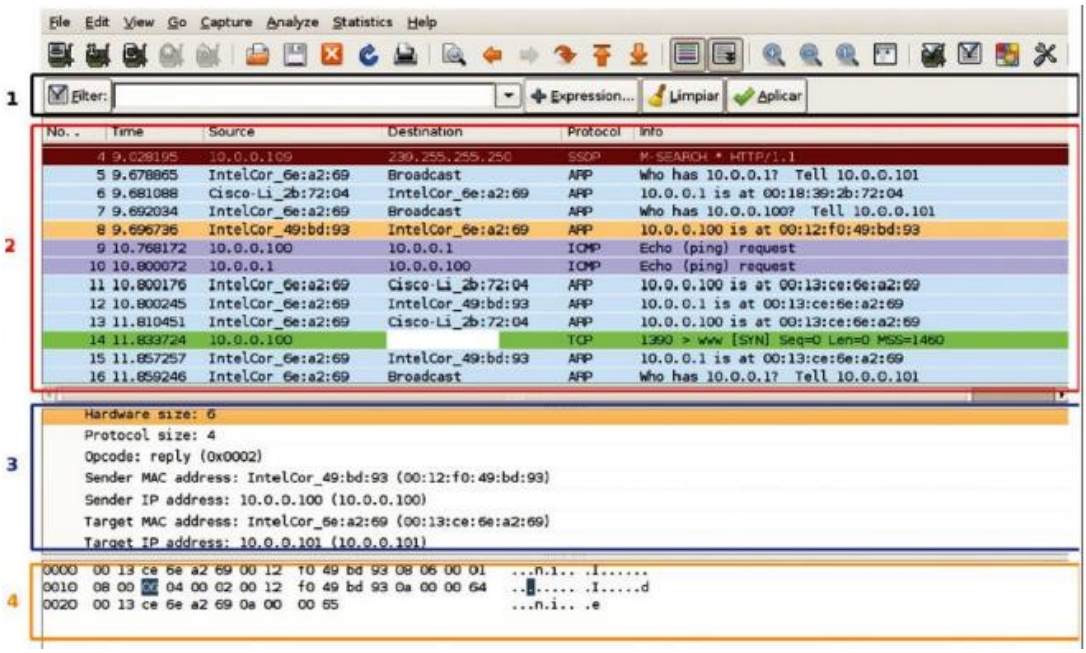


Figura 46.*Áreas de Wireshark*

A continuación, se describe brevemente las áreas más interesantes que nos muestra Wireshark según comienza la toma de datos (Figura 46):

- **La zona 1** es el área de definición de filtros y, como veremos más adelante, permite definir patrones de búsqueda para visualizar aquellos paquetes o protocolos que nos interesen.
- **La zona 2** corresponde a la lista de visualización de todos los paquetes que se están capturando en tiempo real. Saber interpretar correctamente los datos proporcionados en esta zona (tipo de protocolo, números de secuencia, flags, marcas de tiempo, puertos, etc.) nos va a permitir, en ciertas ocasiones, deducir el problema sin tener que realizar una auditoría minuciosa.
- **La zona 3** permite desglosar por capas cada una de las cabeceras de los paquetes seleccionados en la zona 2 y nos facilitará movernos por cada uno de los campos de las mismas.
- Por último, **la zona 4** representa, en formato hexadecimal, el paquete en bruto, es decir, tal y como fue capturado por nuestra tarjeta de red.

2. Como implementar el mecanismo Syn Proxy para la detección y mitigación de ataques DDoS.

Configurar un servidor proxy con Windows Server

2.1. Para comenzar, hacer clic en el menú de Windows en la sección “Control Panel”.

Figura 47. *Panel de control Windows.*

2.2. Luego desde esta ventana dirigirse a “Redes e Internet” y dentro de esta sección, “Opciones de Internet”.

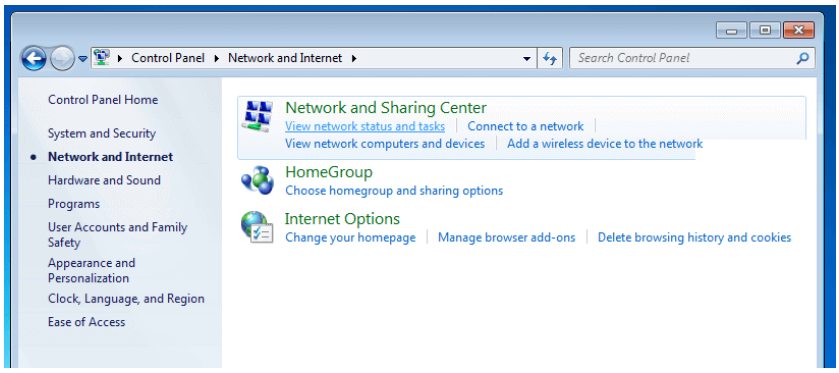


Figura 48. *Redes e Internet Windows.*

2.3. Se abre la ventana “Propiedades de Internet”.

Figura 49. *Propiedades de Internet Windows.*

2.4. En la pestaña “Conexiones” se encuentra el botón “LAN Settings”, que abre, la ventana “Ajustes de la red de área local”.

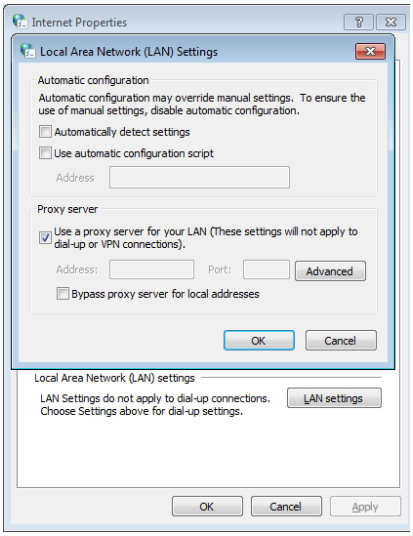


Figura 50. *Ajustes de la red de área local Windows.*

2.5. En este cuadro de diálogo existen tres posibilidades de configuración. Windows puede reconocer una configuración proxy de forma automática, utilizar un script de configuración automática o recurrir a la configuración manual.

2.6. Si el servidor proxy se tiene que configurar manualmente, es recomendable no marcar la casilla del reconocimiento automático de la configuración, pues esta opción podría

superponerse a la configuración manual. En su lugar y como se quiere configurar un servidor proxy se tiene que marcar la tercera opción, donde se introduce la dirección del proxy, así como el número del puerto.

2.7. Haciendo clic en "Avanzado" Dirigirse al cuadro de diálogo "Ajustes proxy", donde se pueden configurar diversos servidores proxy para HTTP, HTTPS, FTP y SOCKS. Si todos los protocolos tienen que utilizar el mismo servidor solo hay que marcar la casilla "Use the same proxy server for all protocols". También es posible definir excepciones ("Exceptions"), de forma que podemos incluir en el campo de texto aquellas direcciones que no han de ser solicitadas con el servidor proxy.

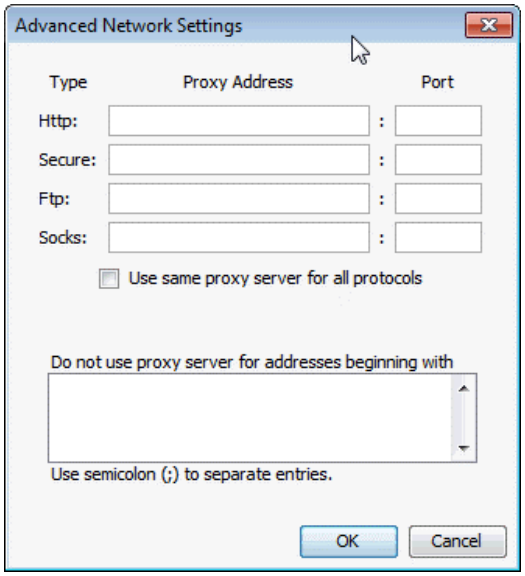


Figura 51. *Configuración Avanzada de red Windows.*

2.8. Los datos que necesitas para la configuración los recibes por parte del proveedor del servidor proxy.

2.9. Algunos servidores están protegidos con datos de acceso individuales. Su acceso requiere en este caso una cuenta de usuario personal con una contraseña.

3. **Como medir el comportamiento del tráfico de nuestra red una vez implementado el mecanismo Syn Proxy.**

En primera instancia, para realizar un análisis dinámico de un código malicioso se procede a infectar un sistema en un entorno controlado. Por lo general, se recurre a

una máquina físicas o virtuales. De esta forma, es posible ejecutar Wireshark y seleccionar la interfaz de red de la máquina virtual para comenzar a capturar los paquetes de red.

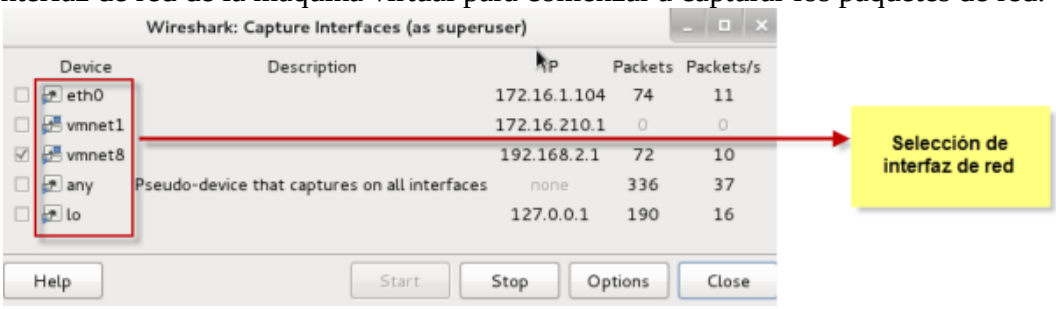


Figura 52. *Selección de interfaz de red.*

3.1.Utilización de Filtros.

Los filtros de captura son aplicados para vigilar paquetes de manera selectiva. Filtrará o capturará únicamente el tráfico definido. Para hacer esta captura selectiva se debe pasar el comando de instrucciones hacia Wireshark.

La opción del filtro de captura puede ser ejecutado haciendo clic en el icono del menú "Edit Capture Filter" o Editar filtro de captura. Se puede también ejecutar haciendo clic en "Capture -> Options" o Captura -> Opciones. Luego de lo cual se visualizará la ventana:

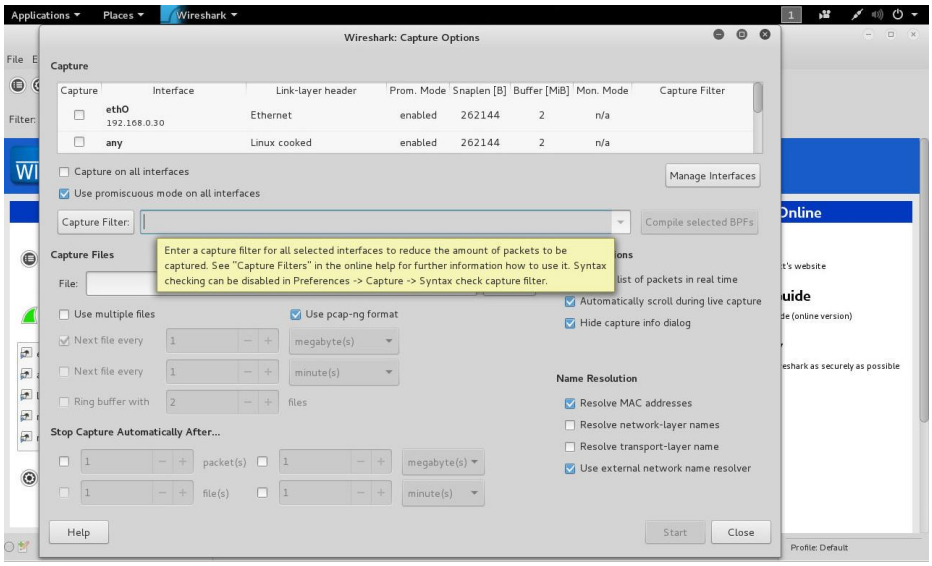


Figura 53. *Filtros de captura.*

Se puede ingresar opciones de filtrado en el campo de nombre “Filtro de Captura”. También es factible hacer clic sobre “Capture Filter” y almacenar la regla de filtro de captura para uso futuro.

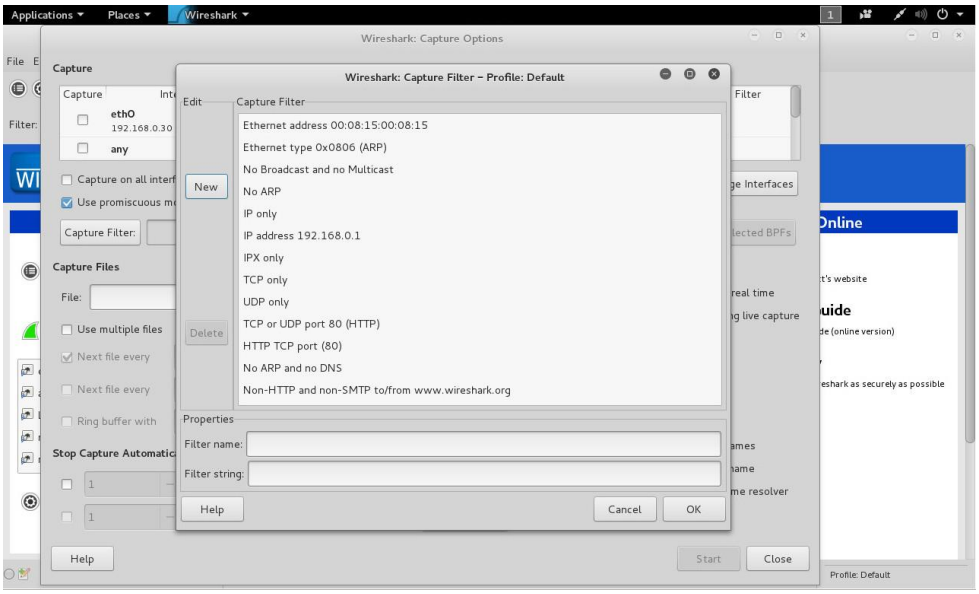


Figura 53*. Filtros de captura.*

Una vez iniciada la captura es también posible filtrar los paquetes capturados de acuerdo a la necesidad de quien esté analizando el malware. Como primera tarea, es posible reconocer a que servidores se conectó a través de las peticiones DNS. Específicamente, si se escribe DNS en el campo de los filtros y se lo aplica, serán visibles todas las resoluciones de nombres en direcciones IP. En el caso del malware, permite reconocer con que servidores se conecta.

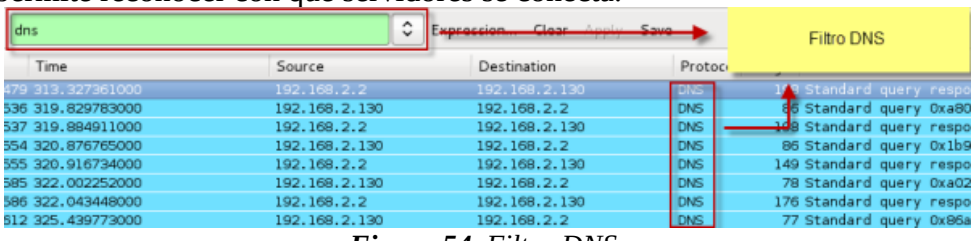


Figura 54. *Filtro DNS.*

Otro aspecto para tener en cuenta son las peticiones realizadas. Aplicando el filtro “http.request” es posible obtener todos los GET y POST que fueron realizados durante el periodo de captura. Este tipo de peticiones es muy utilizado por los códigos

maliciosos, incluso para enviar información sobre el sistema infectado. A continuación, puede observarse una captura de un análisis real sobre un malware que obtiene datos y archivos desde un servidor remoto:

Figura 55. *Peticiones Get y Post.*

Un caso particular han sido aquellos códigos maliciosos que utilizan el protocolo SMTP para propagarse a través de correo electrónico. En estos casos es posible visualizar dichos paquetes a través del filtro SMTP. Esto se realiza a través del filtro "smtp.req.parameter && contains "FROM"". De la misma forma, es posible visualizar aquellos paquetes que contienen el cuerpo del mensaje. Esta tarea puede realizarse a través de filtro "smtp.data.fragment".

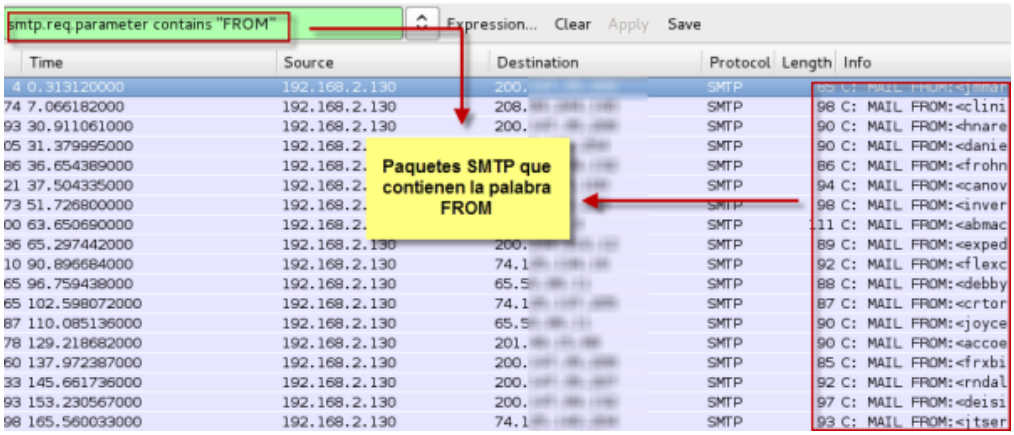
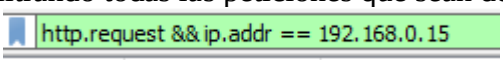


Figura 56. *Paquetes SMTP.*

Además SMTP para filtrar los correos, pueden ser concatenados con operaciones lógicas, para tener un filtro más exacto de lo que se requiere. En la Imagen, por ejemplo, se están filtrando todas las peticiones que sean de la IP 192.168.0.15.



Lista de filtros a utilizar en Wireshark

Combinación de Filtros.

Se puede combinar las primitivas de los filtros de la siguiente forma:

Negación: ! ó not

Unión o Concatenación: && ó and

Alternancia: || ó or

Filtros basados en hosts	
Sintaxis	**Significado**
host host	Filtrar por host
src host host	Capturar por host origen
dst host host	Capturar por host destino
Ejemplos	
host 192.168.1.20	Captura todos los paquetes con origen y destino 192.168.1.20
src host 192.168.1.1	Captura todos los paquetes con origen en host 192.1681.1
dst host 192.168.1.1	Captura todos los paquetes con destino en host 192.168.1.1
dst host SERVER-1	Captura todos los paquetes con destino en host SERVER-1
hosthttp://www.terra.com	Captura todos los paquetes con origen y distinohttp://www.terra.com
Filtros basados en puertos	
Sintaxis	**Significado**
port port	Captura todos los paquetes con puerto origen y destino port
src port port	Captura todos los paquetes con puerto origen port
dst port port	Captura todos los paquetes con puerto destino port
not port port	Captura todos los paquetes excepto origen y destino puerto port
not port port and not port port1	Captura todos los paquetes excepto origen y destino puertos port y port1

Ejemplos	
port 21	Captura todos los paquetes con puerto origen y destino 21
src port 21	Captura todos los paquetes con puerto origen 21
not port 21 and not port 80	Captura todos los paquetes excepto origen y destino puertos 21 y 80
portrange 1-1024	Captura todos los paquetes con puerto origen y destino en un rango de puertos 1 a 1024
dst portrange 1-1024	Captura todos los paquetes con puerto destino en un rango de puertos 1 a 1024
Filtros basados en protocolos Ethernet / IP	
Ejemplos	
ip	Captura todo el tráfico IP
ip proto \tcp	Captura todos los segmentos TCP
ether proto \ip	Captura todo el tráfico IP
ip proto \arp	Captura todo el tráfico ARP
Filtros basados en red	
Sintaxis	**Significado**
net net	Captura todo el tráfico con origen y destino red net
dst net net	Captura todo el tráfico con destino red net
src net net	Captura todo el tráfico con origen red net
Ejemplos	
net 192.168.1.0	Captura todo el tráfico con origen y destino subred 1.0
net 192.168.1.0/24	Captura todo el tráfico para la subred 1.0 mascara 255.0
dst net 192.168.2.0	Captura todo el tráfico con destino para la subred 2.0
net 192.168.2.0 and port 21	Captura todo el tráfico origen y destino puerto 21 en subred 2.0
broadcast	Captura solo el tráfico broadcast
not broadcast and not multicast	Captura todo el tráfico excepto el broadcast y el multicast

Otro punto para tener en cuenta es que a medida que se obtienen los paquetes de acuerdo con el filtro aplicado, es posible obtener toda la secuencia del paquete completo si es necesario. Para ello, solo es necesario colocar el cursor sobre el propio paquete y en el menú contextual elegir la opción de “follow tcp stream”. De esa forma se podrá visualizar el paquete completo tal como se muestra en la siguiente figura: