ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS

INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

PROYECTO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS

TEMA: "ANÁLISIS DEL TRÁFICO DE RED EN LOS LABORATORIOS ESPECIALIZADOS DEL DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN"

ÁREA DE CONOCIMIENTO: GERENCIA ADMINISTRATIVA
LÍNEA DE INVESTIGACIÓN: SEGURIDAD INFORMÁTICA

AUTOR: SAAVEDRA ORTIZ, MARTHA LEONOR
SIMBAÑA GARCÍA, VERÓNICA ELIZABETH

DIRECTOR: ING. ÑACATO, GERMÁN

CODIRECTOR: ING. SOLIS, FERNANDO

SANGOLQUÍ
MAYO, 2015

i

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

CERTIFICADO

Ing. Germán Ñacato (DIRECTOR DE TESIS)
Ing. Fernando Solís (CODIRECTOR DE TESIS)

CERTIFICAN

Que el presente trabajo titulado "ANÁLISIS DEL TRÁFICO DE RED EN LOS LABORATORIOS ESPECIALIZADOS DEL DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN" fue realizado en su totalidad por la Srta. Martha Leonor Saavedra y la Srta. Verónica Elizabeth Simbaña García como requerimiento parcial a la obtención del título de INGENIERO EN SISTEMAS E INFORMÁTICA

Mayo, 2015

ING. GERMÁN ÑACATO
DIRECTOR

ING. FERNANDO SOLIS
CODIRECTOR

ii

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

DECLARACIÓN DE RESPONSABILIDAD

Nosotras, Martha Leonor Saavedra Ortiz y Verónica Elizabeth Simbaña García

DECLARAMOS QUE:

El proyecto de grado denominado "ANÁLISIS DEL TRÁFICO DE RED EN LOS LABORATORIOS ESPECIALIZADOS DEL DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN", ha sido desarrollado con base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas, cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es de nuestra autoría.

En virtud de esta declaración, nos responsabilizamos del contenido veracidad y alcance científico del proyecto de grado en mención.

Sangolquí, Mayo de 2015

Martha Leonor Saavedra Ortiz
C.C: 080265159-6

Verónica Elizabeth Simbaña García
C.C: 172241004-8

iii

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

AUTORIZACIÓN DE PUBLICACIÓN

Nosotras, Martha Leonor Saavedra Ortiz y Verónica Elizabeth Simbaña García, autorizamos a la UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE, la publicación, en la biblioteca virtual de la Institución del proyecto de tesis "ANÁLISIS DEL TRÁFICO DE RED EN LOS LABORATORIOS ESPECIALIZADOS DEL DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN", cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y autoría.

Sangolquí, Mayo de 2015

Martha Leonor Saavedra Ortiz
C.C: 080265159-6

Verónica Elizabeth Simbaña García
C.C: 172241004-8

DEDICATORIA

El presente proyecto va dedicado de manera muy especial a mis padres por su dedicación y amor, por ser mis amigos, consejeros, por siempre guiarme y ser la voz fundamental e incondicional en los momentos difíciles de mi vida profesional, emocional y me ayudaron a culminar esta etapa muy importante en mi vida, a mi hermana a mis tíos que siempre confiaron y me apoyaron.

Verónica Simbaña García

El resultado de este proyecto va dedicado a mis amados padres Martha Ortiz y Guido Saavedra que con su sacrificio y consejos me ayudaron a culminar esta etapa tan importante en mi vida, a mis hermanos que son mi ejemplo a seguir como profesional, a mi abuelita, mis tíos que siempre me apoyaron y confiaron en mí.

Martha Saavedra Ortiz

AGRADECIMIENTO

Agradezco de todo corazón a mis padres, quienes con su incondicional apoyo, por su ejemplo de superación y perseverancia han forjado la senda de mi camino para que pueda realizar mis sueños, y llegar hasta este momento de mi vida a mi hermana quien ha sido un apoyo incondicional en cada momento de mi carrera estudiantil.
A la Universidad de las Fuerzas Armadas, por abrir sus puertas y darme la confianza necesaria para triunfar en la vida, a mis profesores por trasmitirme sabiduría para mi formación profesional. Agradezco de manera muy especial al Ing. Germán Ñacato por su esfuerzo, dedicación, colaboración y al Ing. Fernando Solís por su colaboración y sabiduría para lograr cumplir mi objetivo planteado y ser un profesional de éxito.

Verónica Simbaña García

Agradezco a Dios por guiarme y'''' darme salud para tener las fuerzas de continuar en los momentos difíciles, a mis padres que son mi pilar por guiarme y brindarme su amor incondicional, a mis hermanos por cuidarme, protegerme y siempre estar pendientes de cada paso que doy, a mi Director de Tesis Ing. Germán Ñacato por su paciencia, amistad y apoyo para culminar el presente proyecto, a mi Co-Director Ing. Fernando Solís por su colaboración y por guiarnos correctamente, finalmente a mi mejor amigo que siempre me ánimo para no derrumbarme y seguir adelante con mi objetivo planteado.

Martha Saavedra Ortiz

ÍNDICE DE CONTENIDOS

CERTIFICADO ..... *i*

DECLARACIÓN DE RESPONSABILIDAD ..... *ii*

AUTORIZACIÓN DE PUBLICACIÓN ..... *iii*

DEDICATORIA ..... *iv*

AGRADECIMIENTO ..... *v*

ÍNDICE DE CONTENIDOS ..... *vi*

RESUMEN ..... *xiii*

ABSTRACT ..... *xiv*

CAPÍTULO 1 ..... *1*

1.1 Introducción **1**

1.2 Antecedentes **2**

1.3 Justificación e Importancia **2**

1.4 Objetivos **3**

1.4.1 Objetivo General 3

1.4.2 Objetivos Específicos 4

1.5 Alcance **4**

2. CAPÍTULO 2 ..... *5*

2.1 Análisis de Red **5**

2.1.1 Pesos ponderados 6

2.2 Wireshark **8**

2.3 Modelo OSI **9**

2.4 Modelo TCP/IP **10**

2.5 Comparación Modelo OSI vs Modelo TCP/IP 11

2.6 Comunicación de Host a Host Modelo OSI 15

2.7 Protocolos 16

2.7.1 TCP 17

2.7.1.1 Características del Protocolo TCP 17

2.7.1.2 Estructura de paquetes TCP 18

2.7.1.3 Establecimiento de la conexión 24

2.7.1.4 Terminación de la conexión 27

2.7.2 ARP 28

2.7.2.1 Características Generales ARP 29

2.7.2.2 Funcionamiento 29

2.7.2.3 Ventajas y Desventajas del Protocolo ARP 30

2.7.2.4 Estructura del paquete ARP 31

2.7.3 ICMP 31

2.7.3.1 Estructura de paquetes ICMP 32

2.7.3.2 Tipos de mensajes ICMP 34

2.7.4 UDP 35

2.7.4.1 Características Generales de UDP 35

2.7.4.2 Ventajas y Desventajas del Protocolo UDP 36

2.7.4.3 Estructura del paquete UDP 36

2.7.5 IP 37

2.7.5.1 Estructura de paquetes IPv4 38

2.8 Tipos de Intrusos Informáticos 40

2.9 Metodología. 42

3. CAPÍTULO 3 45

3.1 PROTOCOLO TCP 45

3.1.1 Analizar las comunicaciones normales del TCP 45

3.1.2 Ataque SYN Flooding 54

3.2 PROTOCOLO ARP 58

3.2.1 Análisis del protocolo ARP 59

3.2.2 Analizar Peticiones / Respuestas normales de ARP 60
3.2.3 Envenenamiento ARP 68
3.2.3.1 Condiciones del envenenamiento ARP 68
3.2.3.2 Cómo funciona el envenenamiento ARP 69
3.2.3.3 Métodos de envenenamiento a la caché ARP 70
3.2.3.4 Identificación de intrusos con Wireshark 71

4. CAPÍTULO 4 80

4.1 Protocolo ICMP 80
4.1.1 Analizar tráfico normal del Protocolo ICMP 80
4.1.2 Proceso PING 81
4.1.3 Ataque ICMP Flooding 84

4.2 Protocolo UDP 89
4.2.1 Analizar tráfico normal del Protocolo UDP 89

4.3 Protocolo IP 94

4.4 Suplantación de una Pagina Web 95
4.4.1.1 Clonación de un Sitio Web 96
4.4.1.2 DNS Spoofing 101
4.4.1.3 Robo de Credenciales 105

5. CAPÌTULO 5 107

5.1 CONCLUSIONES 107

5.2 RECOMENDACIONES 109

6. Bibliografía 110

ÍNDICE DE FIGURAS

Figura 1 Correspondencia capas Modelos TCP/IP Y OSI 11
Figura 2 Comunicación de host a host 16
Figura 3 Encabezado TCP 18
Figura 4 Ejemplo número de secuencia 20
Figura 5 Ejemplo número de confirmación 21
Figura 6 Establecimiento de la conexión 26
Figura 7 Terminación de la conexión 28
Figura 8 Diagrama ARP 30
Figura 9 Encabezado ARP 31
Figura 10 Encabezado ICMP 32
Figura 11 Encabezado UDP 37
Figura 12 Formato de un datagrama IPv4. 38
Figura 13: Metodología para la detección de vulnerabilidades 43
Figura 14 Diagrama de la red ESPE 44
Figura 15 Captura de intercambios de Paquetes 46
Figura 16 Comportamiento de las conexiones TCP 47
Figura 17 Detalle de la sección Frame TCP 49
Figura 18 Detalle de la sección Ethernet TCP 51
Figura 19 Detalle del Protocolo IP TCP 53
Figura 20 Detalle de la sección del TCP 54
Figura 21 Ataque TCP/SYN Flooding 55
Figura 22 Ataque con el comando hping3 57
Figura 23 Filtrado del segmento TCP 58
Figura 24 Sin acceso a HTTP 58
Figura 25 Pantallas de Ejemplo de Filtros 60
Figura 26 Captura Trafico ARP 61
Figura 27 Detalle de la sección Frame ARP 62

Figura 28 Detalle de la sección Ethernet ARP....63
Figura 29 Detalle de la sección del Protocolo ARP....64
Figura 30 Captura Respuesta ARP....65
Figura 31 Detalle de la sección Frame respuesta ARP....65
Figura 32 Detalle de la sección Ethernet II respuesta ARP....66
Figura 33 Detalle de la respuesta ARP....67
Figura 34. Envenenamiento ARP....69
Figura 35 Diagrama de Red Ataque ARP....72
Figura 36 Captura Exportar Trama ARP....73
Figura 37 Trama ARP....73
Figura 38 Sustitución MAC de la Víctima....74
Figura 39 Sustitución de la Dirección MAC del Remitente....74
Figura 40 Reemplazo Dirección MAC del atacante....75
Figura 41 Sustitución por la Dirección del Gateway....75
Figura 42 Sustitución por la MAC de la Vctima....76
Figura 43 IP Hexadecimal de la Víctima....76
Figura 44 Envenenamiento cache ARP....77
Figura 45 Consulta y Comprobación del Ataque....78
Figura 46 Verificación envenenamiento ARP....79
Figura 47 Verificar la conexión entre PCs....81
Figura 48 Wireshark Filtrado de Paquetes....82
Figura 49 Detalle de la sección del Frame ICMP....82
Figura 50 Detalle de la sección Ethernet ICMP....83
Figura 51 Detalle de la sección IP ICMP....83
Figura 52 Detalle de la sección ICMP....84
Figura 53 Diagrama de Red Ataque ICMP Flooding....85
Figura 54 Ataque Comando Hping3....86
Figura 55 Ataque ICMP Flooding....87
Figura 56 Descripción del frame ICMP....87

Figura 57 Descripción del detalle del paquete 88
Figura 58 Estado del Rendimiento de la Red 89
Figura 59 Captura tráfico UDP 91
Figura 60 Detalle de la sección Frame DNS 91
Figura 61 Detalle de la sección Ethernet II DNS 92
Figura 62 Detalle de la sección IP DNS 92
Figura 63 Detalle de la sección UDP DNS 93
Figura 64 Detalle de la sección DNS 94
Figura 65 Suplantación de una Página Web 96
Figura 66 Comando Sudo Setoolkid 97
Figura 67 Menú Ataque Ingeniería Social 97
Figura 68 Menú Ataque Web 98
Figura 69 Menú Ataque Harvester 98
Figura 70 Menú Clonar Sitio 99
Figura 71 Suplantación Página Web 99
Figura 72 URL Pagina Clonación 100
Figura 73 Clonación de Página Web 100
Figura 74 Comprobación Sitio Web 101
Figura 75 Comando ettercap 102
Figura 76 Valores DNS Spoofing 102
Figura 77 Comando Ataque ettercap 103
Figura 78 Ataque exitoso 104
Figura 79 Resultado DNS Spoofing 104
Figura 80 Ingreso de Credenciales 105
Figura 81 Captura Usuario y Contraseña 106

ÍNDICE DE TABLAS

Tabla 1. Alternativas de Analizadores 6
Tabla 2. Criterios de Selección 6
Tabla 3: Matriz de Selección de la Herramienta 7
Tabla 4: Matriz de selección del tipo de herramienta 7
Tabla 5: Resultado de selección de tipo de herramienta 7
Tabla 6: Modelo OSI 14
Tabla 7: Modelo TCP/IP 15
Tabla 8 Filtros Protocolo TCP 45
Tabla 9: Descripción de Filtros 59
Tabla 10: Resumen de Datos para el Ataque 72
Tabla 11 Filtros del protocolo ICMP 80
Tabla 12 Filtros del protocolo UDP 90
Tabla 13 Filtros del Protocolo IP 95

RESUMEN

En la actualidad existen ataques a las redes de instituciones, bancos, empresas entre otros, las cuales tienen información preciada y personal que es sensible a que personas malintencionadas quieran invadir y robar la misma. El objetivo del presente proyecto es realizar el análisis de la red utilizando la técnica Sniffing mediante la herramienta Open Source Wireshark, para lo cual se siguió una metodología que permite conocer las potencialidades del programa, inspeccionar el sistema a estudiar y realizar las capturas para luego analizar el tráfico real de los paquetes que viajan a través de la red: tiempo de ingreso del paquete al canal, dirección fuente y destino, longitud promedio de paquete, utilización promedio del canal, etc. Para llevar a cabo nuestro propósito se comenzó a capturar datos en la red y así detectar si existen intrusos para tomar medidas pertinentes para evitar la fuga o manipulación de información y mejorar el rendimiento de la red mediante el monitoreo y el análisis del tráfico, se observó que los estudiantes solo ingresan a páginas de consultas y es muy poco probable que en el momento de capturar el tráfico alguna persona esté realizando ataques por lo que se realizaron simulaciones de los mismos como son SYN Flooding, Envenenamiento ARP, ICMP Flooding y DNS Spoofing. Los beneficios que se obtuvo al realizar el proyecto es que se evidencio que los usuarios son vulnerables y pueden ser víctimas de ataques que pueden dejar al usuario sin servicio de internet, redireccionar información que debe llegar a un usuario o al router hacia la computadora del atacante y así modificar información confidencial.

Palabras Claves

WIRESHARK
REDES INSTITUCIONALES
SPOOFING
FLOODING

ABSTRACT

At present there are attacks on the network of institutions, banks, companies and others, which are precious and personal information that is sensitive to malicious people want to invade and steal it. The objective of this project is to analyze the network using the technique by Sniffing Wireshark Open Source tool, for which a methodology that allows us to know the potential of the program, inspect the system to study and make the catch and then continued analyze real traffic packets traveling through the network: time income channel package, source and destination address, packet length average, average channel utilization, etc. To carry out our purpose began collecting data in the network and test for intruders to take appropriate measures to prevent leakage or manipulation of information and improve the network performance by monitoring and traffic analysis.

It was observed that only students entering pages consultations and it is highly unlikely that at the time of capture traffic attacks someone is doing so simulations are the same as SYN Flooding, Poisoning ARP, ICMP Flooding and DNS are made Spoofing. The benefits to be obtained to carry out the project was evident it is that users are vulnerable and may be victims of attacks that can leave the user without internet service, redirecting information must reach a user or the router to the computer of the attacker and so modify confidential information.

KeyWords

WIRESHARK
NETWORK OF INSTITUTIONS
SPOOFING
FLOODING

CAPÍTULO 1

1.1 Introducción

El amplio desarrollo de las nuevas tecnologías informáticas y la dependencia de sus redes de datos, han dado paso a nuevos retos, siendo uno de los más importantes el de mantener la seguridad de sus sistemas, debido a que los mismos son un punto crítico para el servicio que prestan.

Anteriormente, cuando las redes estaban diseñadas para cubrir solamente una oficina local (redes de área local) el tener simplemente un software antivirus y ciertas medidas físicas de seguridad ayudaba de gran manera a proteger la información de ser destruida o robada por personas que perseguían dichos fines. En los tiempos actuales dicho modelo no sería efectivo ni práctico debido a que las empresas tienen ya por lo menos una puerta abierta hacia Internet, puesto que utilizan dicho servicio para enviar mails, realizar consultas y tener contacto con los clientes.

Ahora bien, para proteger la información en una empresa donde exista una conexión con la red pública se utilizan estrategias de firewalls, routers, NAT, software de firewall, antispyware, detección de intrusos, entre otros. Estos métodos de protección (por experiencia de los investigadores) no ofrecen un 100% de efectividad, por lo cual, siempre existe la posibilidad de que se produzcan fallas en la seguridad.

Esta investigación contempla las vulnerabilidades que puedan existir; así como el tipo de tráfico que viaja en la red a fin de analizar las tramas que circulan en la misma y de esta manera, detectar si transitan contraseñas o algún tipo de información importante sin encriptar, que pueda atentar contra la confidencialidad de los usuarios.

1.2 Antecedentes

Todo administrador de redes en determinada instancia debe enfrentarse a una pérdida de conectividad o de rendimiento en la red que gestiona; en ese caso sabrá que no siempre es sencillo un diagnóstico, ya sea por falta de recursos o por desconocimiento de las herramientas apropiadas.

Históricamente, los analizadores de red se utilizaban en dispositivos de hardware que eran caros y difíciles de usar. Sin embargo, nuevos avances en tecnología han permitido el desarrollo de analizadores de redes basadas en software, lo que hace que sea más conveniente y asequible para los administradores, para así solucionar con eficiencia los problemas de una red ya que gran parte de los problemas están basados en una mala configuración del entorno como puede ser: tormentas broadcast, spanning-tree defectuosos, enlaces redundantes, etc. Y, en determinadas ocasiones y circunstancias, puede tratarse de ataques inducidos por terceros que pretenden una intrusión o dejar fuera de servicio a un servidor mediante un ataque DoS, husmear tráfico mediante un envenenamiento ARP o simplemente infectar algunos equipos con código malicioso. (Zeas Marín, 2011)

A partir del año 2006 Ethereal es conocido como Wireshark, una herramienta gráfica utilizada por los profesionales administradores o usuarios de la red para identificar, analizar y capturar todo tipo de tráfico en un momento determinado.

1.3 Justificación e Importancia

El análisis de redes también conocido como análisis de tráfico, análisis de protocolos, o el análisis de paquetes es el proceso de captura y análisis del tráfico de la red para determinar lo que está sucediendo en la misma.

Un analizador de paquetes decodifica los datos de los protocolos comunes y muestra el tráfico de red en un formato legible.

Un analizador de red puede ser un dispositivo de hardware independiente con software especializado, o software que se instala en un computador de escritorio o portátil. Los analizadores de red dependen de las características tales como el número de protocolos soportados que se pueden descifrar, la interfaz de usuario, gráficos y las capacidades estadísticas.

Entre los beneficios de analizar el tráfico de red se encuentran:

- Mejorar el rendimiento de la red orientando los recursos a fines administrativos, académicos y disminuyendo el uso indebido del internet en los laboratorios como son las redes sociales, pornografía entre otros.
- Detectar si existen intrusos en la red, y de ser así tomar las medidas pertinentes para evitar la fuga o manipulación de información.
- Prevenir la infección de equipos, con código malicioso como son malware.
- Mejorar el rendimiento de la red mediante el monitoreo y análisis del tráfico.

1.4 Objetivos

1.4.1 Objetivo General

Realizar el análisis de red mediante la captura del tráfico utilizando la herramienta open source para determinar posibles problemas de seguridad en los laboratorios especializados del Departamento de Ciencias de la Computación.

1.4.2 Objetivos Específicos

- Utilizar procesos de captura de tráfico mediante sniffer, utilizando la herramienta Open Source Wireshark.
- Capturar el tráfico de la red utilizando la técnica sniffing, que consiste en monitorear todo lo que sucede en una red.
- Interpretar los datos capturados, con el fin de recomendar posibles soluciones a los problemas de alto riesgo.

1.5 Alcance

Mediante el uso de la herramienta Open Source Wireshark se capturará y analizará el tráfico de la red en los laboratorios especializados del Departamento de Ciencias de la Computación basados en los protocolos ARP, TCP, IP, ICMP y UDP.

- En el protocolo ARP se detectará las direcciones duplicadas IPv4.
- En el protocolo TCP se realizará el monitoreo del flujo de los datos.
- En el protocolo IP se analizará la distribución de paquetes de información a su destino, ya que este protocolo no garantiza la recepción del paquete.
- En el protocolo ICMP se analizará errores, alertas y notificaciones generales sobre la red.
- En el protocolo UDP no se utiliza directamente por lo que se analizará mediante el protocolo DNS, el cual hace uso para resolver peticiones de consulta a la mayor brevedad.

2. CAPÍTULO 2

2.1 Análisis de Red

Los analizadores de protocolos de red ("sniffers"), visualizan el tráfico de paquetes que circulan por las redes de computadores, permitiendo analizar el comportamiento de las mismas, detectando errores, congestión, errores de configuración, cuellos de botella, fluctuaciones del tráfico, tormenta de broadcast y tráfico inusual en la red.

Su funcionamiento consiste en capturar una copia de estos paquetes para realizar un análisis posterior, el cual se presenta textual o gráficamente, dependiendo de las capacidades de la herramienta en cuestión como son Wireshark, SmartSniff, Kismet, TCPDump entre otros. (Ruiz, 2015)

A continuación se describe unas 4 herramientas que se podrían utilizar para el desarrollo de la investigación.

- Wireshark: Es un sniffer que monitoriza el tráfico y captura los paquetes de datos que circulan por una determinada red. (Gómez, 2011)
- SmartSniff: Es una herramienta capaz de mostrar el contenido de los paquetes que circulan por una red WIFI. (Gonzalez, 2011)
- Kismet: Es un analizador de tráfico de red que permite capturar los paquetes de red que circulan por la interfaz de red de nuestro equipo. (Barahora & Gellibert, 2011)
- TCPDump: Es una herramienta en línea de comandos cuya utilidad principal es analizar el tráfico que circula por la red. (Jimenez, 2013)

2.1.1 Pesos ponderados

Existen varias herramientas para analizar el tráfico que se listan en la tabla 1 los cuales fueron definidos anteriormente, existen muchos factores y parámetros para seleccionar el tipo de herramienta a utilizar.

Tabla 1. Alternativas de Analizadores

Alternativas	
A	Wireshark
B	SmartSniff
C	Kismet
D	TCPDump

En la tabla 2 se enumeran los criterios y sus respectivas ponderaciones para la seleccionar la herramienta más adecuada.

Tabla 2. Criterios de Selección

Criterios de Selección		
I	Multiplataforma	30 %
II	Interfaz Amigable	40 %
III	Soporte Redes Tipo Wireless o Cable	10 %
IV	Open Source	10 %
V	Soporte	10 %
Total		100%

Para elegir la mejor herramienta, cada opción ha sido calificada con un valor de 1 a 5 siendo, siendo 1 bajo y 5 óptimo, construyendo así la matriz de selección tabla 3.

Tabla 3: Matriz de Selección de la Herramienta

	Criterios de selección				
Alternativas	I	II	III	IV	V
Wireshark	5	5	5	5	1
SmartSniff	1	4	2	5	1
Kismet	1	2	2	1	5
TCPDump	5	2	5	5	1
Total	12	13	14	16	8

Una vez ponderada la matriz, se normaliza la matriz dividiendo para el total cada uno de los criterios de selección. Como se observa en la Tabla 4

Tabla 4: Matriz de selección del tipo de herramienta

	Criterios de selección				
Alternativas	I	II	III	IV	V
Wireshark	0.417	0.385	0.357	0.313	0.125
SmartSniff	0.083	0.307	0.143	0.313	0.125
Kismet	0.083	0.154	0.143	0.061	0.625
TCPDump	0.417	0.154	0.357	0.313	0.125
Total	1	1	1	1	1

Al contar con la matriz normalizada se multiplica cado uno de los criterios por la ponderación, y se suma por cada alternativa de diseño obteniendo como resultado la tabla 5.

Tabla 5: Resultado de selección de tipo de herramienta

	Criterios de selección						
Alternativas	I	II	III	IV	V	∑	%
Wireshark	0.125	0.154	0.036	0.031	0.012	0.358	35.8
SmartSniff	0.025	0.123	0.014	0.031	0.012	0.205	20.5
Kismet	0.025	0.062	0.014	0.006	0.062	0.169	16.9
TCPDump	0.125	0.062	0.036	0.031	0.012	0.266	26.6

Por lo tanto la herramienta más adecuada para realizar el análisis del tráfico de red es la opción de Wireshark ya que tiene las características acorde a los requerimientos de la investigación. (Tufiño, 2012)

2.2 Wireshark

Wireshark es un analizador de tráfico de red, antes conocido como Ethereal, pertenece a la categoría de software libre y es ampliamente utilizado en el ámbito de las redes, realiza el análisis para tomar decisiones que permitan resolver problemas en redes de comunicaciones, Wireshark admite observar el comportamiento de los protocolos en una red, y es una herramienta didáctica para la educación. Además cuenta con todas las características estándar que pueda tener un analizador de protocolos, como son depurar protocolos y aplicaciones de red, capturar diversas tramas, reconocer la trama capturada y mostrar al usuario la información decodificada.

Wireshark permite examinar datos de una red en tiempo real o de un archivo de alguna captura anterior. Se puede analizar la información capturada, a través de los detalles y sumarios por cada paquete. (Barahora & Gellibert, 2011)

Características

- Disponible para UNIX, LINUX, Windows y Mac OS.
- Captura los paquetes directamente desde una interfaz de red.
- Permite obtener detalladamente la información del protocolo utilizado en el paquete capturado.
- Cuenta con la capacidad de importar/exportar los paquetes capturados desde/hacia otros programas.
- Filtra los paquetes que cumplan con un criterio definido previamente.

- Sus funciones gráficas son muy poderosas ya que identifica mediante el uso de colores los paquetes que cumplen con los filtros establecidos.

2.3 Modelo OSI

Es un modelo de siete capas, cada una agrupa algunas de las funciones requeridas para comunicar sistemas, poseen estructura jerárquica. Cada capa se apoya en la anterior, realiza su función y ofrece un servicio a la capa superior.

Ventajas

- Divide la comunicación de red en partes más simples.
- Normaliza los componentes de red y permite el desarrollo por parte de diferentes fabricantes.
- Permite que hardware y software de red diferente, se comunique.
- Los cambios en una capa no afectan las demás.
- Se simplifica el aprendizaje por la división de funciones

Desventajas

- **Mala tecnología:** Las capas no están bien dimensionadas.

- **Mala política:** OSI fue siempre visto como una imposición (no es sugerido como TCP/IP). (Seoane, 2011)

2.4 Modelo TCP/IP

TCP/IP es un conjunto de protocolos que cubren los distintos niveles del modelo OSI. Los dos protocolos más importantes son el TCP (Transmission Control Protocol) y el IP (Internet Protocol), que son los que dan nombre al conjunto.

Es el protocolo común utilizado por todos los ordenadores conectados a Internet, de manera que éstos puedan comunicarse entre sí. Hay que tener en cuenta que en Internet se encuentran conectados ordenadores de clases muy diferentes y con hardware y software incompatibles en muchos casos, además de todos los medios y formas posibles de conexión. Aquí se encuentra una de las grandes ventajas del TCP/IP, pues este protocolo se encargará de que la comunicación entre todos sea posible. TCP/IP es compatible con cualquier sistema operativo y con cualquier tipo de hardware. (Cerda, 2014)

Ventajas

- El conjunto TCP/IP está diseñado para enrutar.
- Tiene un grado muy elevado de fiabilidad.
- Es adecuado para redes grandes y medianas, así como en redes empresariales.
- Se utiliza a nivel mundial para conectarse a Internet y a los servidores web. Es compatible con las herramientas estándar para analizar el funcionamiento de la red. (Salamanca, 2014)

Desventajas

- Es más difícil de configurar y de mantener.
- Es algo más lento en redes con un volumen de tráfico medio bajo. (Perez L. , 2012)

2.5 Comparación Modelo OSI vs Modelo TCP/IP

Como se observa en la Figura 1, ambos modelos tienen mucho en común. Las funcionalidades de los niveles y el concepto de protocolos independientes son muy similares. En el modelo OSI se definen tres conceptos: servicios, interfaces y protocolos.

- Servicios (funciones): Qué hace la capa
- Interfaces: Cómo las capas vecinas pueden solicitar o dar servicios
- Protocolos: Reglas entre capas pares para la comunicación entre sí.

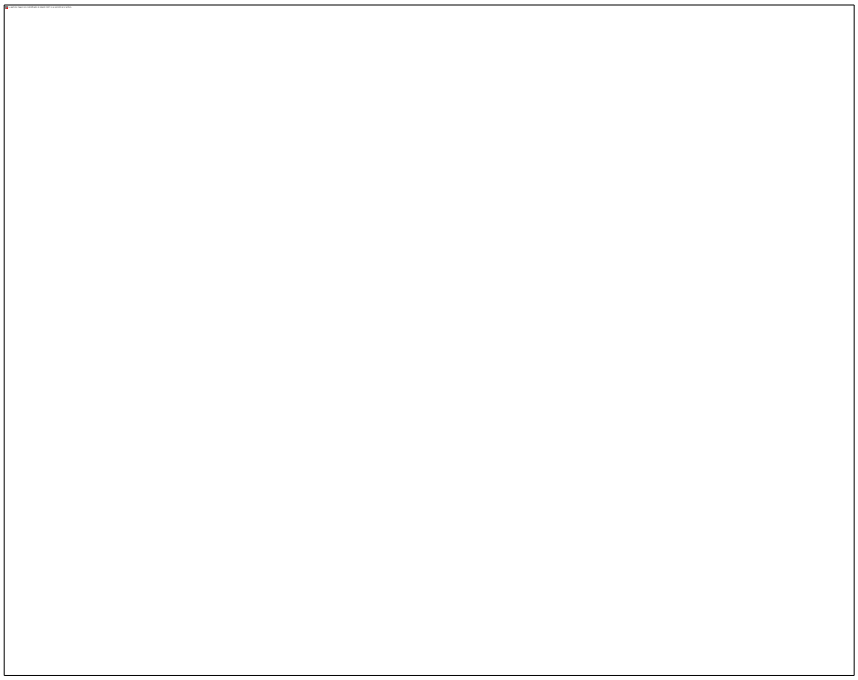


Figura 1 Correspondencia capas Modelos TCP/IP Y OSI

Fuente: (Neto, 2011)

El modelo OSI se desarrolló antes de que se inventaran los protocolos. Aparecieron entonces dificultades para asignar funcionalidades a cada capa, debido a la falta de conocimiento y experiencia de algunos diseñadores. En muchos casos, los protocolos diseñados no cuadraban exactamente en el nivel determinado y se requerían subcapas para tratar esas diferencias

En cambio con TCP/IP, primero llegaron los protocolos y luego se definió el modelo de acuerdo a dichos protocolos. El modelo no se ajusta exactamente a ninguna otra pila de protocolos, esto hizo que no fuera de mucha utilidad para otras redes distintas a las de TCP/IP.

En el modelo de referencia OSI, hay siete capas numeradas, cada una de las cuales ilustra una función de red específica. Las siete capas del modelo de referencia OSI son:

Capa 7: La capa de Aplicación es la capa cercana al usuario; suministra servicios de red a las aplicaciones del usuario. No proporciona servicios a ninguna otra capa OSI, sino solamente a aplicaciones que se encuentran fuera del modelo OSI.

Capa 6: La capa de presentación garantiza que la información que envía la capa de aplicación de un sistema pueda ser leída por la capa de aplicación de otro. De ser necesario, la capa de presentación traduce entre varios formatos de datos utilizando un formato común.

Capa 5: La capa de sesión establece, administra y finaliza las sesiones entre dos hosts que se están comunicando. Proporciona sus servicios a la capa de presentación. También sincroniza el diálogo entre las capas de presentación de los dos hosts y administra su intercambio de datos.

Capa 4: La capa de transporte segmenta los datos originados en el host emisor y los reensambla en una corriente de datos dentro del sistema del host receptor. Proporciona un servicio confiable y se utilizan dispositivos de detección y recuperación de errores de transporte.

Capa 3: La capa de Red es una capa compleja que proporciona conectividad y selección de ruta entre dos sistemas de hosts que pueden estar ubicados en redes geográficamente distintas y se encarga de selección de ruta, direccionamiento y enrutamiento.

Capa 2: La capa de enlace de datos proporciona tránsito de datos confiable a través de un enlace físico.

Capa 1: La capa física define las especificaciones eléctricas, mecánicas, de procedimiento y funcionales para activar, mantener y desactivar el enlace físico entre sistemas finales.

Aunque el modelo de referencia OSI sea universalmente reconocido, el estándar abierto de Internet desde el punto de vista histórico y técnico es el Protocolo de control de transmisión/Protocolo Internet (TCP/IP). El modelo de referencia TCP/IP y la pila de protocolo TCP/IP hacen que sea posible la comunicación entre dos computadores, desde cualquier parte del mundo.

El modelo TCP/IP tiene cuatro capas: la capa de aplicación, la capa de transporte, la capa de Internet y la capa de acceso de red.

La Capa de aplicación contiene las capas de aplicación, presentación y sesión del modelo OSI. Que maneja protocolos de alto nivel, aspectos de representación, codificación y control de diálogo. El modelo TCP/IP combina todos los aspectos relacionados con las aplicaciones en una sola capa y garantiza que estos datos estén correctamente empaquetados para la siguiente capa.

Capa de transporte se refiere a los aspectos de calidad del servicio con respecto a la confiabilidad, el control de flujo y la corrección de errores. Uno de sus protocolos, el

protocolo para el control de la transmisión (TCP), ofrece maneras flexibles y de alta calidad para crear comunicaciones de red confiables, sin problemas de flujo y con un nivel de error bajo.

Capa de Internet se encarga de enviar paquetes origen desde cualquier red en el internet y que estos paquetes lleguen a su destino independientemente de la ruta y de las redes que recorrieron para llegar hasta allí. El protocolo específico que rige esta capa se denomina Protocolo Internet (IP).

Capa de acceso de red es la capa que se ocupa de todos los aspectos que requiere un paquete IP para realizar un enlace físico, incluye los detalles de tecnología LAN y WAN. (Llagua, 2012).

Protocolos Modelo OSI vs TCP/IP

En las siguientes tablas 6 y 7 se mostrará los protocolos que se usan en las diferentes capas de los modelos:

Tabla 6: Modelo OSI

CAPAS	APLICACIONES	PROTOCOLOS
Aplicación	La web, los servicios de correo electrónico, base de datos cliente/servidor	SMTP(Simple Mail Transfer Protocol), TCP/IP
Presentación	Comprime datos	
Sesión	Envío y recepción de un mensaje (mismo conjunto de protocolo)	TCP/IP, IPX/SPX (protocolos orientados a conexión, protocolos sin conexión)
Transporte	Regulación de flujo de mensajes, retransmisión de paquetes.	TCP, SPX,etc.
Red	Enrutamiento de paquetes en la red	IP, IPX, VTAM,etc.
Enlace	Manejo de colisiones,	LAN, Ethernet(IEEE 802.3), Token
Física	Conexión física entre el nodo y la red	RS-232C, RS-449, V24, V35

Fuente: (Chacon, Neto, & Vega, 2011)

Tabla 7: Modelo TCP/IP

Capas	Aplicaciones	Protocolos
Aplicación	Servicios de red, servicios de administración de archivos e impresiones, servicio de conexión a la red.	TCP o UDP
Transporte	Un programa, una tarea, un proceso	TCP (orientado a conexión), UDP (No orientado a conexión)
Internet	Enrutamiento de datagramas	IP, ICMP, ARP, RARP, IGMP
Acceso a la red	Enrutamiento de datos, sincronización, conversión de señal, detección de errores.	ETHERNET, IEEE 802.2, X.25

Fuente: (Chacon, Neto, & Vega, 2011)

2.6 Comunicación de Host a Host Modelo OSI

Cuando un usuario decide enviar un mensaje de correo electrónico a otro usuario de la red. El usuario que envía el mensaje utilizará un cliente o programa de correo (Outlook) como herramienta de interfaz para escribir y enviar el mensaje. Esta actividad del usuario se produce en la capa de aplicación.

Si los datos abandonan la capa de aplicación (la capa insertará un encabezado de capa de aplicación en el paquete de datos), estos pasan par las restantes capas del modelo OSI, cada capa proporcionará servicios específicos relacionados con el enlace de comunicación que debe establecerse, o bien formateará los datos de una determinada forma.

AI margen de la función específica que tenga asignada cada capa, todas adjuntan un encabezado (los encabezados vienen representados por cuadros en la Figura 2) a los datos. Puesto que la capa física está integrada por dispositivos de hardware (un cable, por ejemplo) nunca añade un encabezado a los datos.

Figura 2 Comunicación de host a host

Fuente: (Compostela, 2013).

Los datos llegan así a la capa física de la computadora del destinatario, desplazándose por el entorno físico de la red hasta alcanzar su destino final, el usuario al que iba dirigido el mensaje de correo electrónico.

Los datos se reciben en la capa física de la computadora del destinatario y empiezan a subir por la pila OSI. A medida que los datos van pasando por cada una de las capas, el encabezado pertinente se va suprimiendo de los datos. Cuando los datos finalmente alcanzan la capa de aplicación, el destinatario puede utilizar su cliente de correo electrónico para leer el mensaje que ha recibido. (Rodriguez J. F., 2012)

2.7 Protocolos

Un protocolo es un método estándar que permite la comunicación entre procesos (que potencialmente se ejecutan en diferentes equipos), es decir, es un conjunto de reglas y procedimientos que deben respetarse para el envío y la recepción de datos a través de

una red. Existen diversos protocolos de acuerdo a cómo se espera que sea la comunicación. Algunos protocolos, por ejemplo, se especializarán en el intercambio de archivos (FTP); otros pueden utilizarse simplemente para administrar el estado de la transmisión y los errores (como es el caso de ICMP), (Jean, 2014)

2.7.1 TCP

Transmission Control Protocol (Protocolo de Control de Transmisión) es uno de los protocolos fundamentales del internet. Muchos programas dentro de una red de datos compuesta por computadoras pueden usar TCP para crear conexiones entre ellos a través de las cuales puede enviarse un flujo de datos.

El protocolo garantiza que los datos serán entregados a su destino sin errores y en el mismo orden en que se transmitieron. (Pinzon, 2014)

2.7.1.1 Características del Protocolo TCP

- **Orientado a conexión:** Los sistemas de los dos extremos se sincronizan para controlar el flujo de paquetes y adaptarse a la congestión de la red. Se establece un circuito virtual en cada sentido de la comunicación.

- **Comunicación fiable:** Garantiza la entrega sin errores y en orden.
  - Confirmación de entrega
  - Solicitud de retransmisiones
  - Detección y corrección de errores (checksum en la cabecera TCP)

- **Control de errores:** Si el paquete se recibe correctamente, el receptor envía una confirmación de entrega. Si no, pide la retransmisión.

- **Control de Flujo:** El protocolo define mecanismos para reducir la tasa de transmisión cuando se detectan pérdidas de paquetes y para incrementar la tasa hasta la capacidad máxima cuando dejan de detectarse errores en la comunicación.

2.7.1.2 Estructura de paquetes TCP

Puerto de origen

Este campo es de especial importancia porque es condición necesaria aunque no suficiente para identificar una conexión virtual.

Puerto de destino

Al igual que en el puerto de origen sirve para identificar una conexión virtual.
En la Figura 3 se puede identificar el siguiente encabezado:

Figura 3 Encabezado TCP

Fuente: (Velazquez, 2014)

Número de Secuencia

Este campo es un número bastante grande (32 bits) que tiene varias funciones. Por un lado, identifica unívocamente a cada paquete dentro de una conexión y dentro de un margen de tiempo. Este número permite detectar si un paquete llega duplicado.

El valor del número de secuencia no es aleatorio, es el orden en bytes que ocupan los datos contenidos en el paquete, dentro de una sesión.

El número de secuencia es siempre creciente, por lo que un paquete posterior a otro siempre tendrá un número de secuencia mayor. Esto significa que, una vez que el número de secuencia tome el valor, $2^{32} - 1$ es decir, el mayor valor que se puede representar con 32 bits, el próximo número de secuencia utilizado será 0, dando la vuelta al marcador.

Por ejemplo en la Figura 4, el primer número de secuencia es 0, y el primer paquete contiene 200 bytes de datos. Por tanto, el segundo paquete tendrá número de secuencia 200, este paquete contiene 12345 bytes de datos, por lo que el próximo paquete tendrá como número de secuencia: 200 (número de secuencia anterior) + 12345=12545, así continua la sesión, hasta que un paquete tiene como número de secuencia $2^{32} - 1$, y contiene 300 bytes de datos. El próximo número de secuencia tendría que ser $2^{32} - 1 +$ 300, pero como solo se dispone con 32 bits para representar este número y dando la vuelta al marcador convirtiendo el 2^{32} en 0 por ser el primer bit de secuencia con el que se debe empezar a contar se queda con 299 que sería el número de secuencia del próximo paquete.

El número de secuencia no tiene por qué comenzar en cero, el principal motivo es que si todas las conexiones empiezan en 0, no se podría reutilizar las conexiones.

Figura 4 Ejemplo número de secuencia

Fuente: (Candelas, 2008)

Número de confirmación

Se utiliza para hacer las confirmaciones de recepción, representa un orden de bytes de los datos muy similar al número de secuencia.

Lo que se envía en el número de confirmación es otro número de 32 bytes que indica cuál es el próximo byte que se espera recibir.

Si en una conexión envía un paquete con número de secuencia 1000 y el paquete contiene 550 bytes de datos, el próximo número de secuencia sería 1550, si se quiere confirmar que se ha recibido el paquete con número de secuencia 1000 se tiene que enviar en el paquete de respuesta un número de confirmación 1550. Ver Figura 5

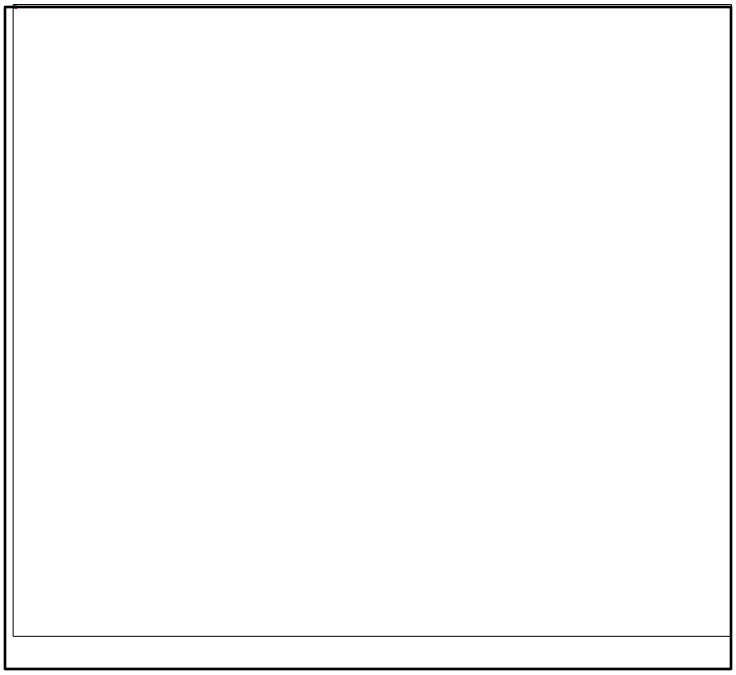


Figura 5 Ejemplo número de confirmación

Fuente: (Candelas, 2008)

Comienzo de datos

La cabecera TCP puede tener un tamaño variable por lo que es necesario indicar a partir de qué punto comienzan los datos y termina la cabecera.
El valor más habitual para este campo es 5 que equivale a 20 bytes de cabecera.

Espacio reservado

Es un espacio reservado de 4 bits, que hay que dejar en cero.

Flags (URG, ACK, PSH, RST, SYN, FIN)

Son una serie de indicadores de control, de un único bit cada uno, con diferentes funciones. Los flags tienen dos estados si está activo vale 1 caso contrario vale 0.

Flag URG

El flag activo indica que el paquete contiene datos urgentes.

Flag ACK

Al estar activo éste flag significa que el paquete aparte de los datos propios, también contiene una confirmación de respuesta a los paquetes que está enviando al otro extremo.

Flag PSH (Push)

Cuando el flag está activo el sistema debe vaciar los buffers de transmisión y recepción. Los buffers son unas colas de paquetes en las que se van almacenando los paquetes que hay que procesar en espera del momento adecuado.

Flag RST (Reset)

Este flag se activa para indicar al otro extremo de la conexión que algo no anda bien, ya que los datos que han llegado no coinciden con la conexión, por lo que se ha perdido la sincronización en ambas partes.

Ante cualquier campo incorrecto que se reciba se tiene que responder con un paquete con este flag activo.

Flag SYN (Sincronización)

Cuando este flag está activo indica al otro extremo que desea establecer una nueva conexión. Al abrir una nueva conexión se utiliza este flag.

Flag FIN (Finish)

Para indicar al otro extremo que la conexión ya se puede cerrar este flag se activa.

Una vez que se envía el flag FIN se tiene que esperar que el otro extremo también envié el suyo para cerrar la conexión.

El flag RST como el FIN se utiliza para finalizar conexiones, con la diferencia que el primero avisa de una situación de error y el segundo avisa de una terminación sin problemas.

Ventana

Este flag se utiliza para llevar a cabo el control de flujo, el cual permite evitar la congestión debida a la diferencia de velocidad entre ambas partes de una conexión.

El tamaño de la ventana está relacionado con la cantidad de espacio libre que se tiene en el buffer de recepción.

Suma de comprobaciones

Se calcula mediante una operación aritmética binaria. Cada vez que se recibe un paquete TCP hay que realizar esta operación y comparar el número obtenido con el campo suma de comprobación del paquete, si ambos no son iguales, los datos son incorrectos y se necesita una retransmisión.

Puntero de urgencia

TCP permite combinar en un mismo paquete datos urgentes con datos no urgentes. El campo puntero de urgencia indica el punto a partir del cual terminan los datos urgentes.

El flag URG debe estar activo para que el campo puntero de urgencia no sea ignorado.

Opciones

Este campo es opcional y es el responsable de que la cabecera TCP sea de tamaño variable. Lo que indica este campo es el máximo tamaño de los segmentos que se está dispuesto recibir. (Garcia A. , 2012)

2.7.1.3 Establecimiento de la conexión

Para establecer una conexión, el TCP utiliza el protocolo three-wayhandshake. Este último necesita tres segmentos TCP para poder establecer la conexión como se puede observar en la Figura 6.

Se considera que el servidor está en un estado de escucha, llamado listen, y que el cliente quiere establecer una conexión con el servidor. El TCP de la máquina cliente iniciará la petición de conexión TCP, que será contestada por el TCP de la máquina servidor.

Para que el cliente TCP pueda establecer una conexión TCP con el servidor, se siguen los pasos siguientes:

1) Petición de la conexión

El TCP cliente envía un segmento de petición de conexión al servidor. Dicho segmento, que se conoce como segmento SYN porque tiene activado el bit SYN en el campo Control de la cabecera del segmento TCP, especifica el número de secuencia inicial TCP del cliente (ISN).

El número de secuencia inicial se elige al azar. La razón es muy sencilla, hay paquetes que pueden sobrevivir en la red una vez se ha cerrado la conexión TCP (incluso si ha sido a causa de una caída del sistema). Es preciso asegurarse de que una conexión nueva elige un número de secuencia inicial que no exista.

Si el sistema cae, pasados unos segundos vuelve a estar en funcionamiento e inmediatamente se establece una conexión nueva utilizando el mismo puerto y la misma dirección IP, se podría interpretar que los segmentos TCP que han quedado retrasados en la red y que ya existían con anterioridad a la caída de la máquina, pertenecen a la conexión nueva, lo que provocaría la confusión y el mal funcionamiento de dicha conexión.

Con el objetivo de protegerse de esta situación, se combinan dos técnicas: una consiste en elegir el número de secuencia inicial de manera aleatoria y la otra es el denominado quiet time, que consiste en que el TCP no crea ninguna conexión nueva después de un rebote de máquinas hasta que no transcurre un tiempo determinado denominado MSL (tiempo máximo de vida de un segmento). De este modo, se asegura de que no recibirá segmentos antiguos de otras conexiones.

2) Confirmación de la conexión

El servidor responde a la petición de establecimiento de la conexión con un segmento SYN que indica el número de secuencia inicial que utilizará. Asimismo, este segmento contiene un reconocimiento (ACK) del segmento SYN del cliente que indica el ISN del cliente más 1 (el número de secuencia inicial del cliente más 1).

3) Reconocimiento de la conexión

Como se observa en la Figura 6, el cliente reconoce el segmento SYN (K) del servidor con un reconocimiento que contiene el ISN servidor más 1. Sería el segmento ACK (K + 1).

Se dice que quien envía el primer segmento SYN (en este caso, el cliente) efectúa una apertura activa (active open), mientras que quien recibe el primer segmento SYN y envía el próximo segmento SYN (en este caso, el servidor) lleva a cabo una apertura pasiva (passive open). (Callejas, 2010)

Puede darse el caso de que ambos extremos efectúen una apertura activa en el mismo momento. Esta situación se denomina apertura simultánea (simultaneous open).

Después de estos tres pasos, se puede decir que ya se ha establecido la conexión entre el cliente y el servidor.

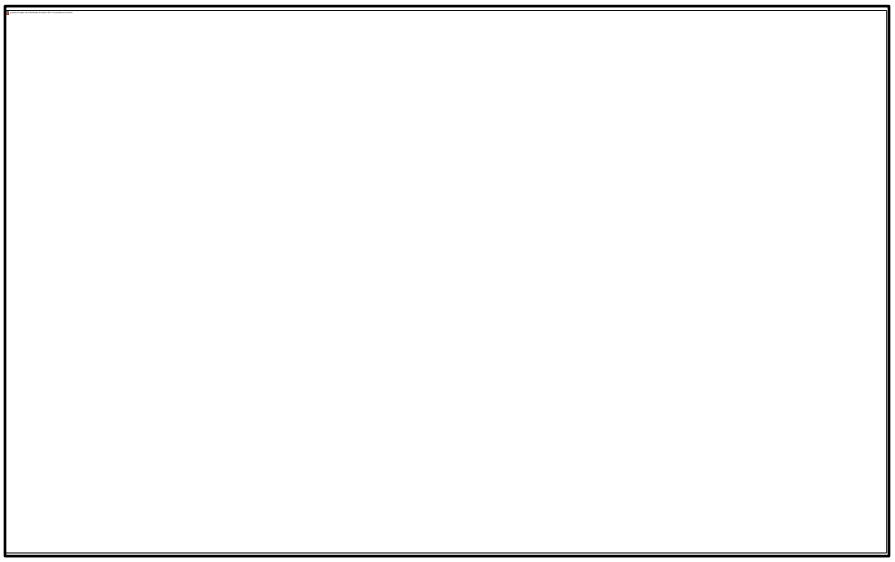


Figura 6 Establecimiento de la conexión

Fuente: (Callejas, 2010)

2.7.1.4 Terminación de la conexión

Cuando la transferencia de la información ha finalizado, el TCP dispone de un protocolo de terminación de la conexión para cerrarla.

En una conexión TCP full duplex, en la que los datos fluyen en ambos sentidos, cualquier conexión debe cerrarse independientemente.

Es preciso tener en cuenta que tanto el cliente como el servidor pueden cerrar la conexión. Sin embargo, la situación normal es que la aplicación cliente inicie la petición de conexión, como se puede observar en la Figura 7. Los pasos que se siguen para cerrar la conexión son los siguientes:

1) El cliente envía un segmento TCP del tipo FIN con el número de secuencia correspondiente (J). Ello significa que a partir de este momento no habrá más datos que fluyan en este sentido (cliente → servidor).

2) El servidor envía una confirmación del cierre por medio de un ACK con el número de secuencia recibido más 1 (J + 1). El TCP servidor indica a su aplicación que el cliente cierra la conexión. La aplicación servidor indica a su TCP que la cierre a continuación.

3) El servidor envía un segmento TCP del tipo FIN al cliente con el número de secuencia correspondiente (K).

4) El TCP cliente responde automáticamente con un ACK (K + 1).

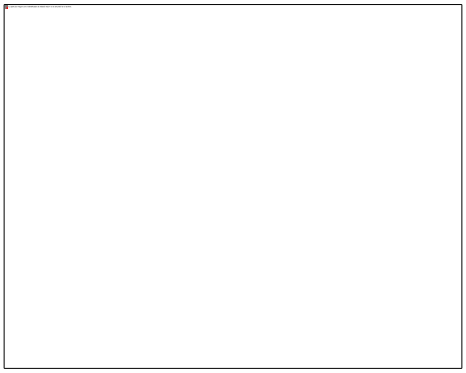


Figura 7 Terminación de la conexión

Fuente: (Callejas, 2010)

Si el cliente envía el primer segmento FIN lleva a cabo un cierre activo (active close), mientras que quien lo recibe, es decir el Servidor realiza un cierre pasivo (passive close).

Es posible que sólo cierre la conexión uno de los extremos, mientras que el otro se mantiene abierto. Esta situación se denomina half-close. Asimismo, puede darse el caso de que dos extremos efectúen un cierre activo. Esta situación se denomina cierre simultáneo. (Verdejo Alvarez, 2012)

2.7.2 ARP

El propósito del protocolo ARP es poder obtener la dirección física (dirección MAC de 48 bits) de un computador, dada su dirección lógica (dirección IP de 32 bits). Este protocolo puede ser explicado en los dos usos que tiene. El primero es cuando se desea hacer la consulta de una dirección MAC entonces se genera una trama de consulta ARP, y el segundo es cuando se recibe una trama ARP ya sea consulta o respuesta.

El protocolo se optimiza con la utilización de cachés ARP. En estas cachés se guarda la correspondencia entre direcciones IP y las direcciones físicas de los nodos de

la red. Antes de enviar una consulta ARP, se trata de resolver la dirección buscándola en las entradas de la caché. (Ordóñez, 2012)

2.7.2.1 Características Generales ARP

- ARP es un protocolo de bajo nivel que oculta direccionamiento de la red en las capas inferiores, permitiendo asignar al administrador de la red direcciones IP a los host pertenecientes a una red física.
- Las tablas ARP son fundamentales para el funcionamiento y rendimiento óptimo de una red, pues reducen el tráfico en la misma al enviar preguntas ARP innecesariamente.
- El protocolo ARP puede ser usado por un posible atacante para objetivos no deseados.

2.7.2.2 Funcionamiento

Una vez que un paquete llega a una red local mediante el ruteo IP, el encaminamiento necesario para la entrega del mismo al host destino se debe realizar forzosamente mediante la dirección MAC del mismo (número de la tarjeta de red).

El protocolo ARP equipará direcciones IP con direcciones Ethernet (de 48 bits) de forma dinámica, evitando así el uso de tablas de conversión.

El protocolo ARP manda a las demás máquinas de su red un mensaje para preguntar qué dirección local pertenece la dirección IP, siendo contestada por una respuesta ARP.

Una vez que la máquina solicitante tiene este dato envía los paquetes al host usando la dirección física obtenida.

Obteniendo ya la dirección con la información se guarda en una tabla de orígenes y destinos de ARP de tal forma que en los próximos envíos ya no habrá que preguntar la

dirección del destinatario porque ya es conocida como se observa en la Figura 8. (Teldat, 2010)

Figura 8 Diagrama ARP

Fuente: (Moreno, 2008)

2.7.2.3 Ventajas y Desventajas del Protocolo ARP

Ventajas

La principal ventaja del uso de la técnica ARP Proxy es que se puede agregar a un solo enrutador en la red, esto permite que no se distorsione las tablas de encaminamiento de los otros enrutadores de la red. Es recomendable que el ARP Proxy sea utilizado en redes donde los host IP no se encuentren configurados con ninguna puerta de enlace predeterminada.

Desventajas

- Aumenta la cantidad de trafico ARP en su segmento
- La seguridad puede ser expuesta. Un host puede simular ser otro host con el fin de interceptar los paquetes, esto es llamado “spoofing”
- No funciona para redes que no utilicen el protocolo ARP para la resolución de direcciones. (A.M, 2013)

2.7.2.4 Estructura del paquete ARP

En la Figura 9 se puede identificar el siguiente encabezado:

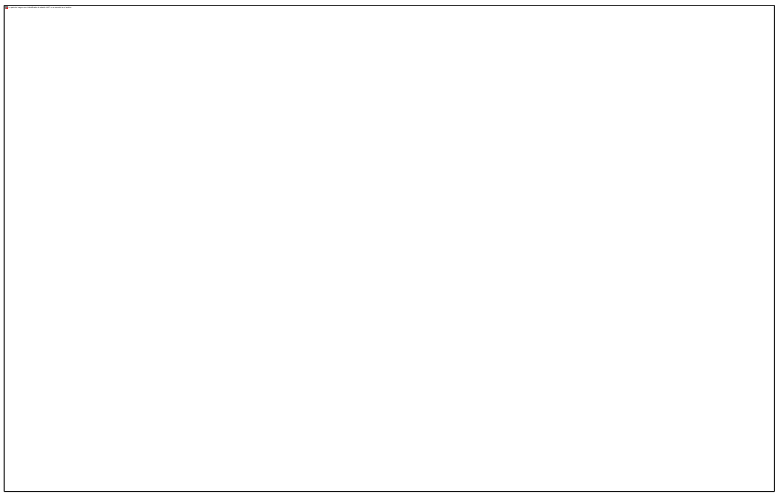


Figura 9 Encabezado ARP

Fuente: (Corletti Estrada, 2011)

2.7.3 ICMP

(Protocolo de mensajes de control de Internet) es un protocolo que permite administrar información relacionada con errores de los equipos en red. ICMP no permite corregir los errores sino que los notifica a los protocolos de capas cercanas. Por lo tanto, el protocolo ICMP es usado por todos los routers para indicar un error (llamado un problema de entrega).

Los mensajes de error ICMP se envían a través de la red en forma de datagramas, como cualquier otro dato. Por lo tanto, los mismos mensajes de error pueden contener errores.

Sin embargo, si existe un error en un datagrama que lleva un mensaje ICMP, no se envía ningún mensaje de error para evitar el tráfico innecesario, si hay un incidente en la red. (Kioskea, 2015)

2.7.3.1 Estructura de paquetes ICMP

Los mensajes ICMP son enviados usando el encabezado básico IP. El primer byte de la porción de datos del datagrama es un campo de tipo ICMP, el valor de este campo determina el formato de los datos restantes. El formato del encabezado IP para ICMP se observa en la Figura 10.

Figura 10 Encabezado ICMP

Fuente: (Gil Vasquez, 2011)

- **Versión:** lleva el registro de la versión del protocolo al que pertenece el mensaje en nuestro caso es 4.
- **IHL (Internet Header Length):** ya que la longitud de la cabecera no es constante, se incluye este campo para indicar la longitud en palabras de 32 bits. El valor mínimo es de 5, cifra que aplica cuando no hay opciones en el encabezado.
- **Tipo de Servicio:** permite al host indicar a la subred el tipo de servicio que quiere, en nuestro caso el tipo es 0.

- **Longitud Total:** incluye la longitud total del mensaje, tanto en la cabecera como en los datos.
- **Identificación:** usado en la fragmentación del mensaje, todos los fragmentos de un mensaje tienen el mismo identificador.

A continuación se tiene un bit sin uso y luego dos campos de 1 bit.

- **DF** (d): (Don´t Fragment) es una orden para los enrutadores de que no fragmenten el mensaje porque el destino es incapaz de juntar las piezas de nuevo.
- **MF** (m): (More Fragments) significa más fragmentos. Todos los fragmentos excepto el último tienen establecido este bit, que es necesario para saber cuándo han llegado todos los fragmentos de un mensaje.
- **Desplazamiento de Fragmento:** indica en que parte del mensaje va este fragmento.
- **Tiempo de Vida:** es un contador que sirve para limitar la vida de un mensaje. Se supone que este contador cuenta el tiempo en segundo, permitiendo una vida máxima de 255 segundos; debe disminuirse en cada salto y se supone que se disminuye muchas veces al encolarse durante un tiempo grande en un enrutador. En la práctica, simplemente cuenta los saltos. Cuando el contador llega acero, el paquete se descarta y se envía de regreso un paquete de aviso al host de origen.
- **Protocolo:** ICMP = 1 (Definido en el RFC 1700)
- **Suma de Comprobación de la Cabecera:** verifica solamente la cabecera y debe recalcularse en cada salto.
- **Dirección de Origen:** La dirección del gateway o host que compuso el mensaje ICMP.
- **Dirección Destino:** La dirección del gateway o host a quien el mensaje va dirigido. (Gil Vasquez, 2011)

2.7.3.2 Tipos de mensajes ICMP

- **Mensajes informativos**

Entre estos mensajes hay algunos de suma importancia, como los mensajes de petición de ECO (tipo 8) y los de respuesta de Eco (tipo 0). Las peticiones y respuestas de eco se usan en redes para comprobar si existe una comunicación entre dos host a nivel de capa de red, por lo que sirve para identificar fallos en este nivel, ya que verifican si las capas física (cableado), de enlace de datos (tarjeta de red) y red (configuración IP) se encuentran en buen estado y configuración.

- **Mensajes de error**

En el caso de obtener un mensaje ICMP de destino inalcanzable, con campo "tipo" de valor 3, el error concreto que se ha producido vendrá dado por el valor del campo "código".

Este tipo de mensajes se generan cuando el tiempo de vida del datagrama ha llegado a cero mientras se encontraba en tránsito hacia el host destino (código=0), o porque, habiendo llegado al destino, el tiempo de reensamblado de los diferentes fragmentos expira antes de que lleguen todos los necesarios (código=1).

Los mensajes ICMP de tipo= 12 (problemas de parámetros) se originan por ejemplo cuando existe información inconsistente en alguno de los campos del datagrama, que hace que sea imposible procesar el mismo correctamente, cuando se envían datagramas de tamaño incorrecto o cuando falta algún campo obligatorio.

Por su parte, los mensajes de tipo=5 (mensajes de redirección) se suelen enviar cuando, existiendo dos o más routers diferentes en la misma red, el paquete se envía al router equivocado. En este caso, el router receptor devuelve el datagrama al host origen

junto con un mensaje ICMP de redirección, lo que hará que éste actualice su tabla de enrutamiento y envíe el paquete al siguiente router. (De La Cruz, 2012)

2.7.4 UDP

UDP es un protocolo no orientado a conexión. Es decir cuando una maquina A envía paquetes a una maquina B, el flujo es unidireccional. La transferencia de datos es realizada sin haber realizado previamente una conexión con la máquina de destino (maquina B), y el destinatario recibirá los datos sin enviar una confirmación al emisor (la maquina A). (Yazid, 2012)

2.7.4.1 Características Generales de UDP

- No es orientado a la conexión.
- No garantiza la fiabilidad
- Hace lo que puede para transmitir los datos hacia la aplicación.
- No preserva la secuencia de la información que proporciona la aplicación, llega con retardos y la aplicación que lo recibe debe estar preparada por si se pierden los datos.
- No envía un mensaje al dispositivo transmisor de que el mensaje se ha recibido en forma correcta.
- Es muy rápida y fácil de utilizar.
- Cuando detecta un error en el dato en lugar de enviarlo a su destino lo elimina.
- Es más sencilla que el TCP ocasiona una interfaz con el IP u otros protocolos sin la molestia del control de flujo de errores, actuando tan solo con un transmisor y receptor de datagramas. (Garcia & Casillas, 2015)

2.7.4.2 Ventajas y Desventajas del Protocolo UDP

Ventajas

- No te restringe a un modelo de comunicación basado en la conexión, la latencia para el inicio en aplicaciones distribuidas es mucho menor, al igual que la sobrecarga del sistema operativo.
- Todo el control de flujo, los acuses de recibo, el registro de transacciones, etc. depende de los programas de usuario. Además, sólo es necesario implementar y utilizar las funciones que necesita.
- El receptor de los paquetes UDP los recibe sin fragmentar, incluyendo los límites de los bloques.
- Broadcast y transmisión multicast están disponibles con UDP.

Desventajas

- No hay garantías con UDP. un paquete puede no ser entregado, o entregado dos veces o entregado fuera de orden, no se obtiene ningún indicio de esto a menos que el programa de escucha en el otro extremo decide decir algo.
- UDP no tiene control de flujo
- Los routers son muy descuidados con UDP, nunca se retransmiten si colisionan, y parecen ser la primera cosa descartada cuando un router está corto de memoria. UDP sufre más pérdida de paquetes que TCP. (Rodriguez L. , 2013)

2.7.4.3 Estructura del paquete UDP

UDP utiliza el protocolo IP para transportar sus mensajes. No añade mejora alguna en la calidad de la transferencia, aunque si incorpora los puertos de origen y destino en su formato de mensaje, el cual se muestra a continuación en la Figura 11:

Figura 11 Encabezado UDP

Fuente: (Systems, 2013)

El significado de cada uno de los campos es el siguiente:

- **Puerto UDP de origen (16 bits):** identifica porque puerto ha enviado la aplicación emisora el paquete y tiene el valor 0 si no se usa.
- **Puerto UDP de destino (16 bits):** identifica el proceso (puerto) que recibirá la información en el nodo de destino.
- **Longitud del mensaje UDP (16 bits):** indica la longitud total del paquete.
- **Suma de verificación UDP (16 bits, opcional):** utilizado para verificar el contenido del paquete. (Systems, 2013)

2.7.5 IP

El protocolo de IP (Internet Protocol) es la base fundamental de la Internet. Porta datagramas de la fuente al destino. El nivel de transporte parte el flujo de datos en datagramas. Durante su transmisión se puede partir un datagrama en fragmentos que se montan de nuevo en el destino. Las principales características de este protocolo son:

- Protocolo orientado a no conexión.
- Fragmenta paquetes si es necesario.
- Direccionamiento mediante direcciones lógicas IP de 32 bits.

- Si un paquete no es recibido, este permanecerá en la red durante un tiempo finito.
- Realiza el "mejor esfuerzo" para la distribución de paquetes.
- Tamaño máximo del paquete de 65635 bytes. (Graciano, 2014)

El propósito principal del protocolo IP es proveer una dirección única dentro de una infraestructura para asegurar que cualquier dispositivo de comunicación o equipo pueda ser identificado en la red.

2.7.5.1 Estructura de paquetes IPv4

IP define el formato que los paquetes deben tener y el modo de utilizarlos durante el envío y la recepción. El formato que toma el paquete se denomina datagrama IP. Los datagramas IP son análogos a las tramas físicas que se transmiten en una red. El formato de un datagrama IPv4 se muestra en la Figura 12. El tamaño normal de un encabezado IP es de 20 bytes, a menos que presente el campo de opciones.

Figura 12 Formato de un datagrama IPv4.

Fuente: (Vives, 2011)

Los campos del datagrama IP se describen a continuación:

- **Ver:** es de 4 bits y contiene la versión del protocolo IP que se utilizó para crear el datagrama.
- **Hlen:** es un campo de 4 bits, que proporciona la longitud del encabezado del datagrama medida en palabras de 32 bits.
- **Tipo de servicio**: es un campo de 8 bits que está subdividido en 5 campos, tres bits para especificar la prioridad del datagrama, los siguientes tres D, T y R especifican el tipo de transporte deseado para el datagrama, y los dos últimos no se utilizan.
- **Longitud total:** proporciona la longitud del datagrama medido en bytes, incluyendo los bytes del encabezado y los datos.
- **Identificación**: contiene un entero único para identificar el datagrama.
- **Banderas:** es un campo de tres bits que controlan la fragmentación, el primer bit no se utiliza, y el segundo es llamado DF que quiere decir no fragmentación y el tercero MF que significa más fragmentos.
- **Desplazamiento de fragmento:** especifica el desplazamiento en el datagrama original de los datos que se están acarreando en el fragmento.
- **Tiempo de vida**: específica la duración en segundos del tiempo que el datagrama tiene permitido permanecer en la red.
- **Protocolo:** contiene un valor que especifica qué protocolo se utilizó para crear el mensaje que se está transportando en el área de datos.
- **Suma de verificación del encabezado**: asegura la integridad de los valores del encabezado.
- **Dirección IP de la fuente y dirección IP del destino**: contienen la dirección IP del emisor y del receptor respectivamente.
- **Opciones:** se incluye en principio para pruebas de red o depuración. (Reyes, 2014)

2.8 Tipos de Intrusos Informáticos

- **Hackers**

Los hackers son intrusos que entran en los sistemas informáticos para demostrar y poner a prueba su inteligencia y conocimientos de internet, pero no pretenden provocar daños en estos sistemas.

Sin embargo, pueden tener acceso a información confidencial, por lo que su actividad está siendo considerada como un delito en bastantes países de nuestro entorno.

En la actualidad muchos "hackers" defienden sus actuaciones alegando que sólo pretenden mejorar y poner a prueba sus conocimientos.

Por otra parte, la actividad de un "hacker" podría provocar otros daños en el sistema que lograrían ser aprovechadas por otros usuarios maliciosos. Además, la organización debe dedicar tiempo y recursos para detectar y recuperar los sistemas que han sido comprometidos por un "hacker". (Sanchez, 2014)

- **Crackers ("blackhats")**

Los crackers son individuos con interés en atacar un sistema informático para obtener beneficios de forma ilegal o simplemente para provocar algún daño a la organización, motivados por intereses económicos, políticos, religiosos, etcétera. (Monroy, 2013)

- **Sniffers**

Los sniffers son individuos que se dedican a rastrear y tratar de recomponer y descifrar los mensajes que circulan por redes de ordenadores como Internet. (Brassfield, 2012)

- **Phreakers**

Los phreakers son intrusos especializados en sabotear las redes telefónicas para poder realizar llamadas gratuitas. (Ubenga, 2011)

- **Spammers**

Los spammers son los responsables del envío masivo de miles de mensajes de correo electrónico no solicitados a través de redes como Internet, provocando el colapso de los servidores y la sobrecarga de los buzones de correo de los usuarios.

Además, muchos de estos mensajes de correo pueden contener código dañino (virus informáticos) o forman parte de intentos de estafa realizados a través de Internet (los famosos casos de "phishing"). (Monroy, 2013)

- **Piratas informáticos**

Los piratas informáticos son los individuos especializados en el pirateo de programas y contenidos digitales, infringiendo la legislación sobre propiedad intelectual. (Monroy, 2013)

- **Creadores de virus y programas dañinos**

Se trata de expertos informáticos que pretenden demostrar sus conocimientos construyendo virus y otros programas dañinos, que distribuyen hoy en día a través de

Internet para conseguir una propagación exponencial y alcanzar así una mayor notoriedad.

En estos últimos años, además, han refinado sus técnicas para desarrollar virus con una clara actividad delictiva. (Brassfield, 2012)

- **Lamers (“wannabes”): “Script kiddies” o “Click-kiddies”**

Los “lamers”, también conocidos por “script kiddies” o “click kiddies”, son aquellas personas que han obtenido determinados programas o herramientas para realizar ataques informáticos (descargándolos generalmente desde algún servidor de Internet) y que los utilizan sin tener conocimientos técnicos de cómo funcionan. (Perez B. , 2011)

2.9 Metodología.

La metodología para el ANÁLISIS DEL TRÁFICO DE RED EN LOS LABORATORIOS ESPECIALIZADOS DEL DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN, que se propone, se basa en conocimientos experimental –práctico que se adquirió mediante la investigación bibliográfica y la experiencia que se obtuvo a lo largo del proceso. Dicha metodología pretende ser explícita la cual pueda servir de referencia en investigaciones siguientes. Para dicho objetivo, se diseñó un proyecto con 3 fases, como se puede observar en la Figura 13.

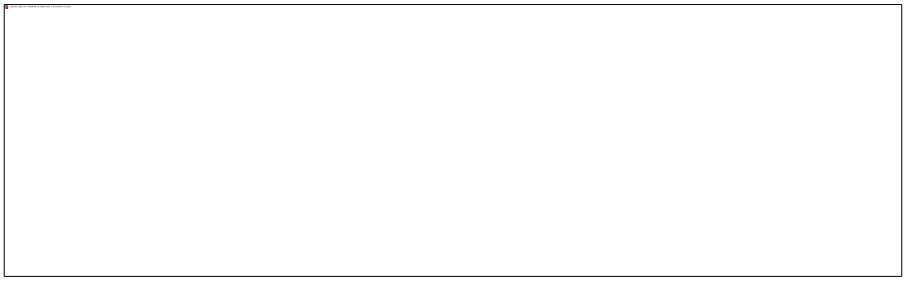


Figura 13: Metodología para la detección de vulnerabilidades

Fuente: (Franco, Perea, & Puello, 2011)

La primera fase consiste en obtener información de la red, y adquirir los conocimientos necesarios tanto de la herramienta Open Source, como de los protocolos y topología de red, Se resalta que esta fase no busca obtener vulnerabilidad alguna, lo que se pretende con ella es obtener la información necesaria de la red y del software a usar. La información recopilada en la primera fase es utilizada en la segunda fase llamada Captura y simulación, en esta fase se realiza una recopilación bibliográfica de los distintos tipos de ataques, además de cómo llevarlos a cabo, para así capturar el tráfico que será usado en la tercera fase llamada Detección de vulnerabilidades donde se plantea métodos para identificar los ataques. (Franco, Perea, & Puello, 2011)

2.9.1. Fase I: Reconocimiento

Esta fase se identificaron los equipos con los que cuenta la red ver Figura 14, los sistemas operativos que manejan, así como la topología de red que existe en los Laboratorios, además se recopiló información de los protocolos del Modelo TCP/IP. Se estudió la herramienta Open Source Wireshark, para efectuar la captura de tráfico, identificar los paquetes deseados y el uso de las distintas formas de visualizar los datos.

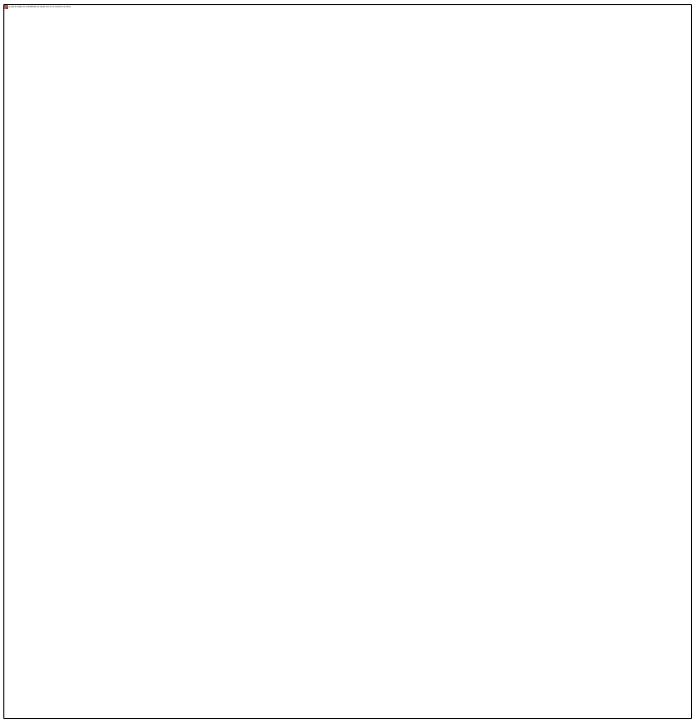


Figura 14 Diagrama de la red ESPE

Fuente: Escuela Politécnica del Ejercito

2.9.2. Fase II: Captura y Simulación

Se determinó los protocolos a estudiar, los cuales se usaron para efectuar la simulación de los ataques, para luego capturar y filtrar por cada uno de los protocolos.

2.9.3. Fase III: Detección de vulnerabilidades

En esta última fase se realizó un análisis de la información mostrada por parte de la herramienta Wireshark para identificar ataques, intrusos y posibles problemas de seguridad existentes en la red.

3. CAPÍTULO 3

3.1 PROTOCOLO TCP

Es el protocolo común utilizado por todos los ordenadores conectados a Internet, de manera que éstos puedan comunicarse entre sí. Se debe tener en cuenta que en Internet se encuentran conectados ordenadores de clases muy diferentes y con hardware y software incompatibles en muchos casos, además de todos los medios y formas posibles de conexión. (Cerda, 2014)

3.1.1 Analizar las comunicaciones normales del TCP

Se puede construir filtros de visualización que comparan valores usando un número de diferentes operadores de comparación.

Los principales filtros utilizados en el protocolo TCP se muestran en la Tabla 8 son:

Tabla 8 Filtros Protocolo TCP

Filtro	Descripción
Tcp.data **Tcp.segment_data**	Datos del segmento TCP
Tcp.dstport	Puerto de Destino TCP
Tcp.flags	Banderas
Tcp.flags.ack	Reconocimiento
Tcp.hdr_len	Tamaño de la Cabecera TCP
Tcp.srcport	Puerto de Origen TCP
Tcp.segment	Segmento TCP

Las comunicaciones TCP se realizan mediante el mecanismo de negociación de tres vías pero antes de comenzar es necesario que los buffers tanto del servidor como el cliente se encuentre vacío, para lo cual existe un intercambio de paquetes ACK con la bandera PUSH activa, como se muestra en la Figura 15.

Figura 15 Captura de intercambios de Paquetes

Se establece una conexión entre dos máquinas, la pc1 con IP 10.9.8.24 envía una solicitud SYN a la pc2 con IP 10.9.8.26, a la cual responde con un paquete SYN/ACK, y por último la pc1 envía un acuse de recibo ACK como se observa en los paquetes 1065 a 1067 de la Figura 15.

Una vez conectadas las dos máquinas se comienza la transmisión de los datos, lo cual se observa en los paquetes 1068 a 1080, finalmente para cerrar la comunicación la PC1 envía un paquete con la bandera FIN activa, luego la PC2 responde con un acuse de recibo ACK y un FIN activo como se muestra en los paquetes 1081 al 1083.

Se puede ver de forma gráfica la secuencia de paquetes seleccionando en el menú Statistics>>FlowGraph. Esta herramienta facilita en numerosas ocasiones seguir el comportamiento de conexiones TCP, como se observa en la Figura 16 donde mediante flechas se describe el origen y destino de cada paquete, resaltando los flags activos que intervienen en cada sentido de la conexión.

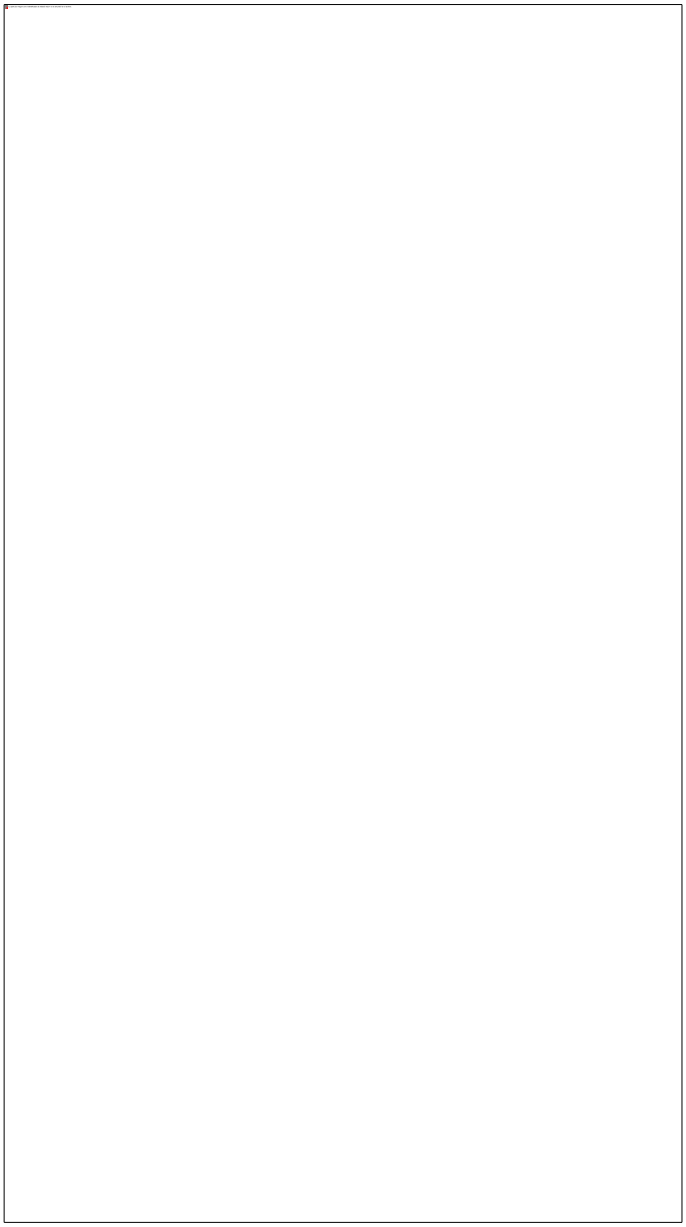


Figura 16 Comportamiento de las conexiones TCP

En la Figura 16 se observa un diagrama de flujos del comienzo de la conexión TCP. Se puede ver el acuerdo “3 way handshake” en las primeras líneas, para luego dar comienzo a la comunicación y envío de paquetes y cómo termina la comunicación TCP entre 2 hosts.

Una de las características que entrega Wireshark es el Throughput de la conexión. Este término hace referencia a la capacidad de un enlace para transportar información útil.

En las siguientes imágenes se explica el detalle del paquete de comunicación TCP de la Figura 16.

La primera sección Frame 1079: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) corresponde a la capa física del modelo OSI, este campo indica que es el frame 1079 de la captura realizada, y que su tamaño es de 54 bytes. En la Figura 17 se observa la siguiente información:

Arrival Time: Nov 10, 2014 19:10:45.636954000 SA Pacific Standard Time: Indica la fecha y la hora en la que se realiza la captura de los datos.

Epoch Time: 1415664645.989348000seconds: Indica la fecha y hora pero en formato UNIX time, el cual es el tiempo medido en segundos desde la media noche del 1 de enero de 1970.

Time delta from previous captured frame: 0.000045000 seconds: Es el tiempo transcurrido desde la captura del último paquete.

Time delta from previous displayed frame: 0.000045000 seconds: Indica el tiempo transcurrido desde que se mostró en pantalla el último paquete capturado.

Time since reference or first frame: 60.635618000 seconds: Indica el tiempo transcurrido desde que se capturo el primer paquete.

Frame Number: 1079: Es el número del paquete capturado.

Frame Length: 54 bytes (432 bits): Indica que la longitud del paquete es de 54 bytes.
Capture Length: 54 bytes (432 bits): Indica que el tamaño de los bytes es 54 bytes.

Frame is marked: False: Indica si el paquete fue marcado por el usuario dentro de la herramienta Wireshark con propósitos de análisis.

Frame is ignored: False: Indica si el paquete fue ignorado por el usuario dentro de la herramienta Wireshark con propósitos de análisis.

Protocols in frame: eth:ethertype:ip:tcp: Indica que el paquete contiene los protocolos ip y tcp.

Coloring Rule Name: TCP: Indica que se ha aplicado una regla de color de paquetes llamada TCP como mecanismo de identificación visual para propósitos de análisis.

Coloring Rule String: tcp: Indica que la regla de color se aplicó al protocolo TCP.

Figura 17 Detalle de la sección Frame TCP

La segunda sección Ethernet II corresponde a la capa de enlace del modelo OSI, pero en el modelo TCP/IP se une las capas 1 y 2 del modelo OSI, en la Figura 18 se observa lo siguiente:

Destination: 00:19:99:57:85:70 (00:19:99:57:85:70): Indica la dirección MAC del destinatario del paquete.

Address: 00:19:99:57:85:70 (00:19:99:57:85:70): Indica la dirección MAC de origen.

.... ..0. .... = LG bit: Globally unique address (factory default): El bit LG indica si la configuración que se utiliza es por defecto o si se modifico y en este caso es igual a 0 lo cual indica que se utilizó la configuración por defecto caso contrario si es 1 la configuración se modifico.

....0 .... = IG bit: Individual address (unicast): El bit IG indica si el usuario de destino o fuente es un solo equipo o todos los de la red y en este casoes igual a 0 es decir que es un solo equipo (unicast) y por lo contrario si fuera 1 serian todos los equipos de la red (broadcast).

Source: e8:9a:8f:1a:1e:93 (e8:9a:8f:1a:1e:9)
Address: e8:9a:8f:1a:1e:93 (e8:9a:8f:1a:1e:9): Indica la dirección MAC de origen.
Type: IP (0x0800): Es el tipo de protocolo que se utilizó.

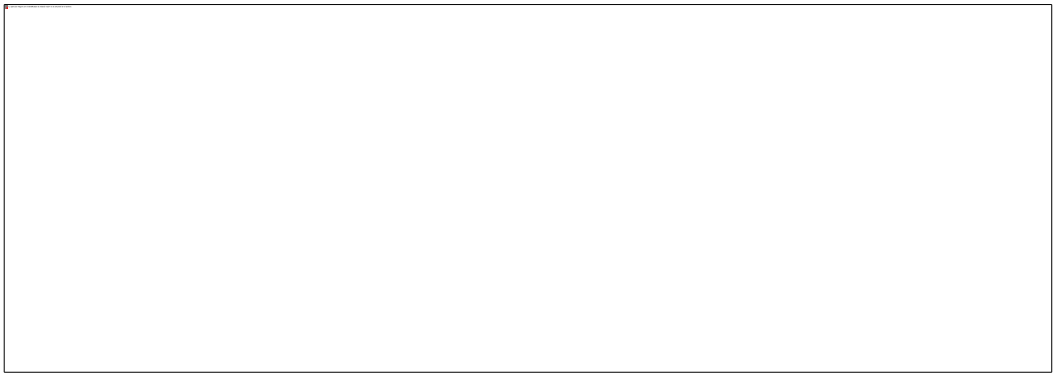


Figura 18 Detalle de la sección Ethernet TCP

La tercera sección corresponde a la capa de red del modelo OSI, en la Figura 19 se observa la siguiente información:

Versión: 4: Indica el formato de la cabecera utilizada.

Header Length: 20 bytes: Este campo describe la longitud de la cabecera en palabras de 32 bits. Su valor mínimo es de 5 para una cabecera correcta, y el máximo de 15. En este caso es 5 y multiplicado por 32 bits da como resultado 160 bits o 20 bytes.

Differentiated Services Field: Tipo de servicio respecto a la fiabilidad, velocidad, retardo, seguridad de la red.

Total Length: 40: Longitud total del datagrama.

Identification: 0x3826 (14374): Es el número de identificación único por cada datagrama que permite el reensamblaje posterior al ser dividido en fragmentos más pequeños. Longitud 16 bits.

Flags 0x002 (Don’t Fragment): Son indicadores de control, usado en caso de desfragmentación.

0… …. = Reserved bit: Not set: El primer bit está reservado y es siempre 0.

.1.. …. = **Don’t fragment: Set**: El segundo es el bit de indicación de no fragmentación.

..0. …. = **More fragments: NOT set:** El tercer bit verifica que el datagrama llega a su destino, está activo en todos los datagramas enviados excepto en el último para informar que ya no hay más fragmentos.

Fragment offset: 0: Posición del fragmento dentro del datagrama en caso de fragmentación, su longitud es de 13 bits.

Time to live: 128: Impide que un paquete esté indefinidamente viajando por la red. En este caso 128 indica que cada vez que un datagrama atraviese un router este número se decrementa en 1 y cuando el TTL llegue a 0 el datagrama se descarta y se informa de ello al origen con un mensaje de tiempo excedido, su longitud es de8 bits.

Protocol: TCP (6): Se refiere al protocolo de siguiente nivel que se usa en la parte de datos, su longitud es de8 bits.

Header checksum: Se indica la suma de comprobación de errores de la cabecera del datagrama, este número se calcula nuevamente en cada salto del datagrama a través de los routers y su longitud es 16 bits.

Source: 10.9.8.126 (10.9.8.126): Dirección de origen y su longitud es 32 bits.

Destination: 10.9.8.124 (10.9.8.124): Dirección de destino y su longitud es 32 bits.

Figura 19 Detalle del Protocolo IP TCP

La cuarta sección corresponde a la capa de transporte que coincide en los dos modelos OSI y TCP/IP. En la figura 20 se observa la siguiente información:

Source Port: 50000 (50000): Es el puerto de origen que representa el protocolo TCP.

Destination Port: 8186 (8186): Puerto de Destino.

Sequence number: 6: Indica el número de secuencia del segmento.

Acknowledgment number: 10279: Indica el número de secuencia del byte que se espera recibir en el siguiente paquete que en este caso es 10279 y dice al otro extremo de la conexión que los bytes anteriores se han recibido correctamente.

Flags: Esta sección tiene activado 1 en el indicador Acknowledgment: Set lo cual significa que hay una comunicación establecida con el otro extremo.

Windows size value: 256: Se indica el número de bytes que el emisor del segmento está dispuesto a aceptar por parte del destino en este caso es 256.

Urgent Pointer: 0: Se utiliza cuando se están enviando datos urgentes que tienen preferencia.

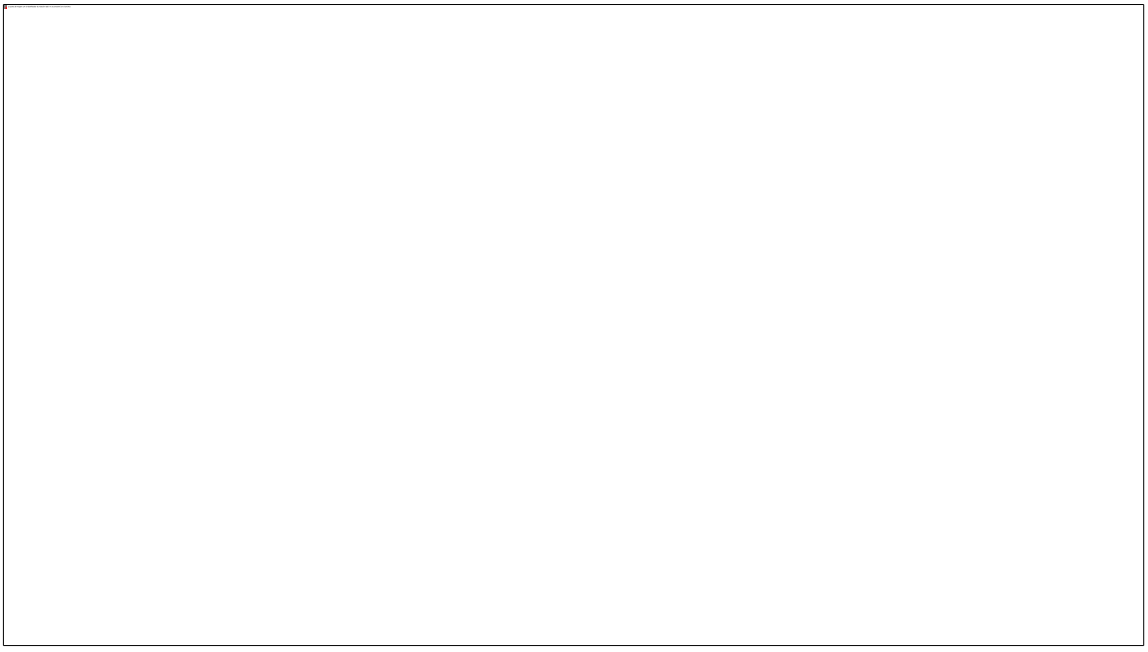


Figura 20 Detalle de la sección del TCP

3.1.2 Ataque SYN Flooding

El ataque TCP/SYN Flooding, se aprovecha del mecanismo de negociación de tres vías, al enviar paquetes SYN para inundar la cola de espera de la víctima, ya que esta se queda esperando por establecer un conexión pues el atacante no responde con ACK los SYN/ACK, esto ocurre hasta saturar los recursos de memoria para así conseguir la denegación de servicios de la víctima.

El atacante para cubrir sus rastros y no ser identificado debe usar un IP falsa.

Para analizar este tipo de ataque se realizó un DoS entre dos computadores del Laboratorio de Multimedia, en el cual el atacante denegará el servicio Http de la víctima mediante un SYN Flooding.

A continuación se muestra la Figura 21 con la información del ataque TCP/SYN Flooding.

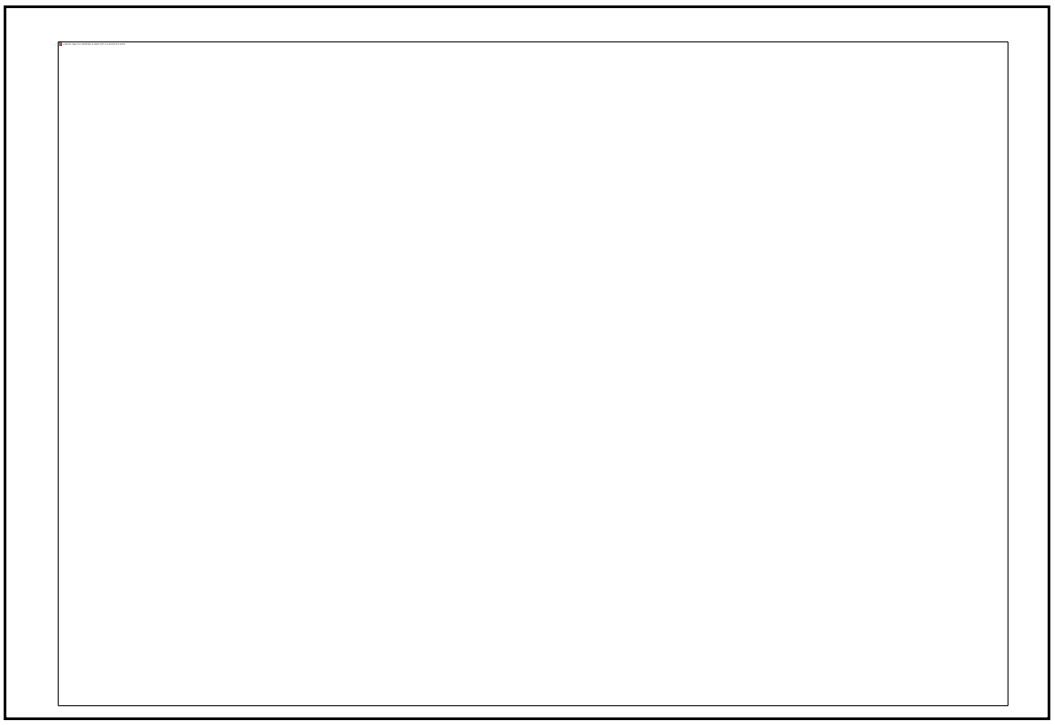


Figura 21 Ataque TCP/SYN Flooding

Se utiliza la IP de Google la cual es 74.125.225.50 para así tratar de que el ataque pase desapercibido por parte de la víctima.

Una vez que el atacante conoce la dirección IP de la víctima 10.9.8.126, es necesario contar con el programa hping3 que genera paquetes TCP SYN a medida.

Programa Hping3

Hping3 es un programa analizador/ensamblador de paquetes TCP/IP de uso en modo consola. Está inspirado en el comando ping de UNIX y puede generar paquetes TPC SYN desde un origen falso.

Hping3 es un programa analizador/ensamblador de paquetes TCP/IP de uso en modo consola. Está inspirado en el comando ping de UNIX y puede generar paquetes TPC SYN desde un origen falso.

Instalación de Hping3

La instalación de esta herramienta en Ubuntu se realiza en la consola al escribir el comando:

root@vero:/home/root# apt-get install hping3

Sintaxis:

La sintaxis para utilizar Hping3 es la siguiente:

hping3 -opción IP

Al utilizar el comando sin ninguna opción actuará igual que el ping.

Algunas de las opciones que tiene son:

-i: especifica el intervalo con el cual hping3 efectúa ping.
--fast: Envía 10 paquetes por segundo.
--faster: Similar a --fast, pero con esta opción se mandan más paquetes de los que el mismo ordenador puede enviar.

--flood: Envía paquetes lo más rápido posible, sin tener en cuenta las respuestas entrantes.

-q: No muestra nada, excepto las líneas de resumen al comenzar y al terminar.

-I: Permite elegir la interfaz (tarjeta de red) por la cual se efectuará el envío de pings.

--rawip: modo IP RAW, en este modo hping3 enviará encabezado IP con datos adjuntos con - firma y / o - file

--icmp: modo de ICMP, por defecto hping3 enviará ICMP

--upd: modo UDP, hping3 enviará paquetes UDP.

--scan: modo de exploración, se añade un rango de puertos para que explore

-a: se utiliza para establecer una conexión con una IP falsa.

Al contar con la herramienta y la IP de la víctima se realiza el ataque con el comando: sudo hping3 –a 74.125.225.50 10.9.8.126 -$ --flood como se muestra en la Figura 22, para identificar el ataque con Wireshark, se observa que existen una gran cantidad de segmentos TCP con el flag SYN activado desde la misma IP y respuestas SYN-ACK por parte de la víctima pero no existe un solo acuse de recibo ACK.

0

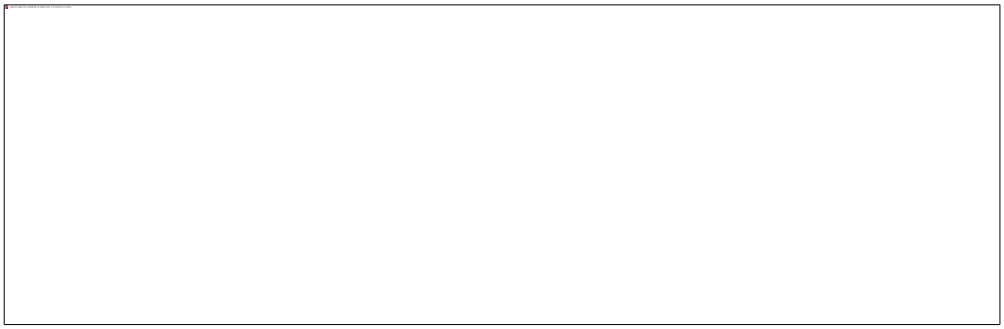


Figura 22 Ataque con el comando hping3

En un intervalo muy corto de tiempo, existen numerosos intentos de conexión por parte de la IP 74.125.225.50 al puerto 80 de la máquina 10.9.8.126 como se observa en la Figura 23, situación algo inusual.

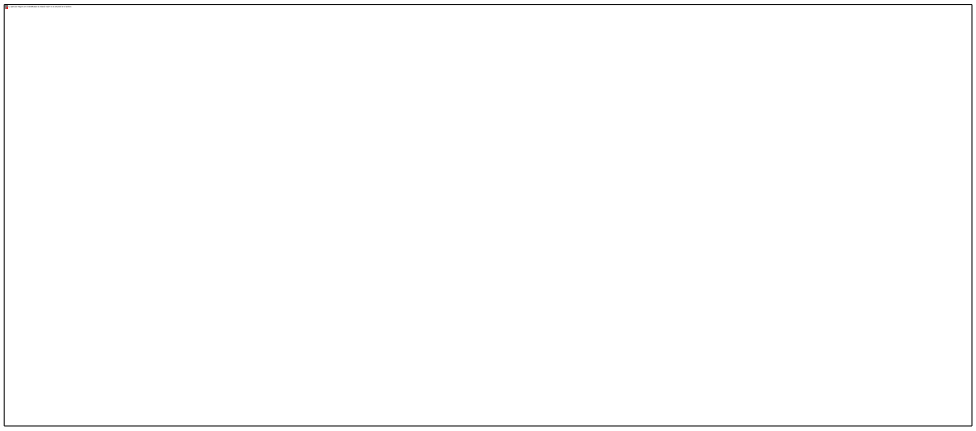


Figura 23 Filtrado del segmento TCP

Y aunque la víctima responde con un paquete SYN-ACK el servidor no confirma respuesta alguna, manteniendo así en modo de espera a la víctima lo que provoca que no cuente con acceso a HTTP mientras dura el ataque. Ver Figura 24.

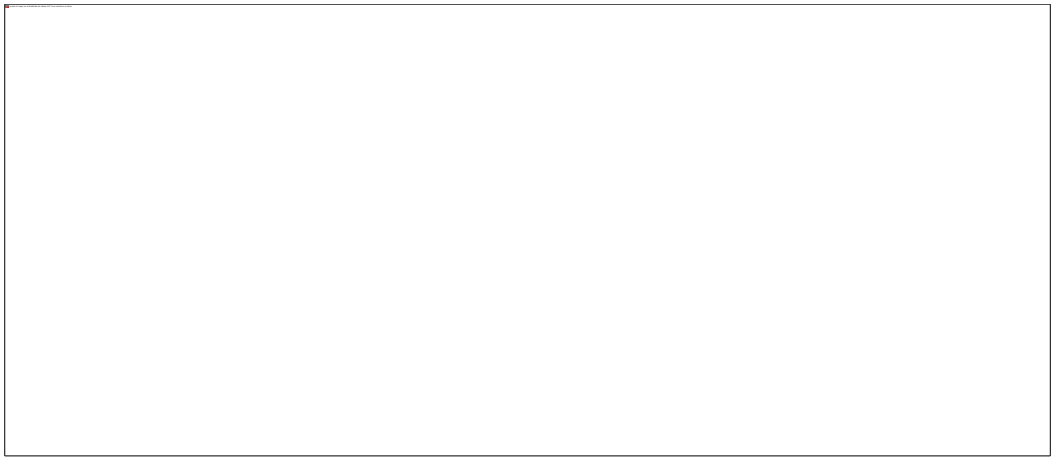


Figura 24 Sin acceso a HTTP

3.2 PROTOCOLO ARP

El protocolo ARP puede obtener la dirección física (dirección MAC de 48 bits) de un computador, dada su dirección lógica (dirección IP de 32 bits). Este protocolo puede ser explicado en los dos usos que tiene. El primero es cuando se desea hacer la consulta

de una dirección MAC entonces se genera una trama de consulta ARP, y el segundo es cuando se recibe una trama ARP ya sea consulta o respuesta. (Ordóñez, 2012)

3.2.1 Análisis del protocolo ARP

Se puede construir filtros de visualización que comparan valores usando un número de diferentes operadores de comparación como se muestra en la Tabla 9.

Tabla 9: Descripción de Filtros

Filtro	Descripción
Arp	Indica solicitudes y respuestas. Ver Figura 3.11ª
arp.opcode == 0x0001	Filtra exclusivamente las solicitudes enviadas por los host. Ver Figura 3.11b
arp.opcode == 0x0002	Muestras las respuestas. Ver Figura 3.11c
arp.src.hw_mac== MAC	Muestras las solicitudes y respuestas de la dirección física señalada. Ver Figura 3.11d
Arp.proto.size	Tamaño del protocolo
Arp.proto.type	Tipo de Protocolo
Arp.hw.type	Tipo de Hardware
Arp.src.hw	Dirección de hardware del remitente
Arp.src.pln	Tamaño protocolo del remitente
Arp.src.proto	Dirección del protocolo remitente
Arp.src.proto_ipv4	Dirección IP del remitente
Arp.hw.size	Tamaño del hardware

En la Figura 25 se muestra un ejemplo de los filtros antes mencionados.

Figura 25 Pantallas de Ejemplo de Filtros

3.2.2 Analizar Peticiones / Respuestas normales de ARP

Al estar analizando el tráfico ARP, deben estar en el mismo segmento de red como host para enviar paquetes ARP y hacer la captura de los mismos.

Las comunicaciones normales ARP consisten en una solicitud y una respuesta simple. Un host envía una difusión de ARP, que incluye la dirección IP de destino (pero no la dirección de hardware de destino que es lo que ARP resuelve).

En el laboratorio se realizó la captura del tráfico ARP para su posterior análisis, en la Figura 26 se verifica que hay una tormenta de broadcast, es decir que se están enviando solicitudes ARP para obtener una respuesta de la dirección física de ciertas IP.

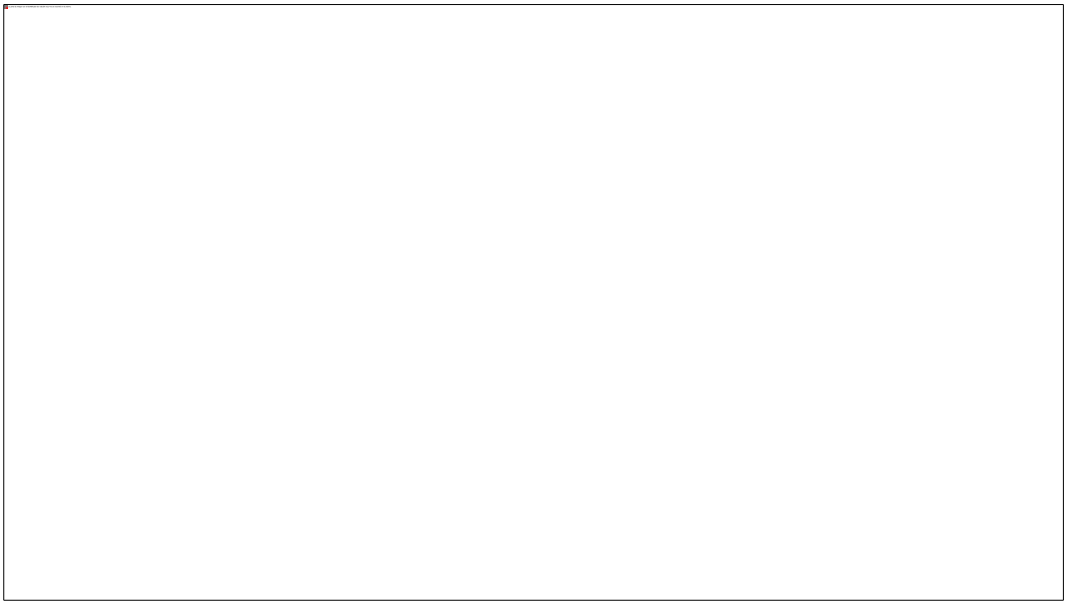


Figura 26 Captura Trafico ARP

Se efectuó el estudio de la solicitud del paquete 195 y se determinó que un host con la dirección de hardware 00:19:99:57:85:70 y la dirección IP 10.9.8.124 solicita cuál es la dirección MAC asociada a la IP 10.9.8.126. Como se observa en el detalle de la trama el campo de dirección de hardware de destino se establece en 0 para indicar que la información no se conoce.

En la Figura 27 se tiene la primera sección que es el Frame, el cual muestra que el paquete es el 195 y que su tamaño es de 60 bytes. Toda la información que se encuentra dentro del Frame es parecida al protocolo TCP pero cambian el contenido de los campos por lo cual no se va a entrar en detalle.

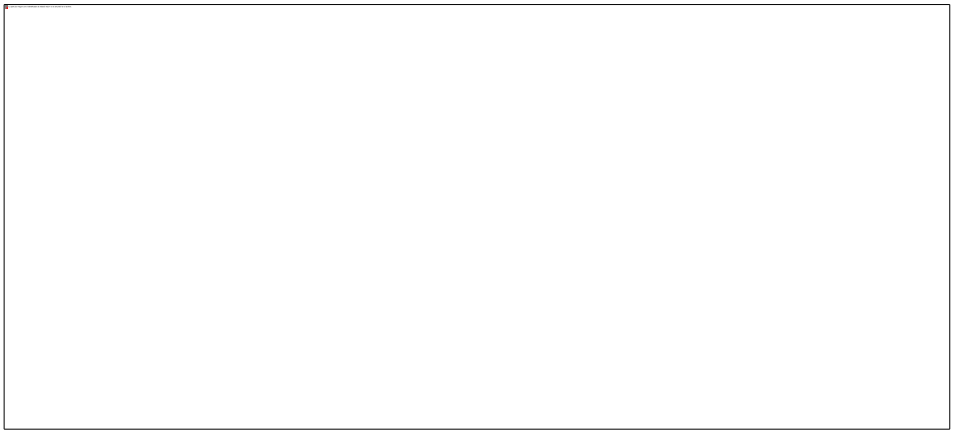


Figura 27 Detalle de la sección Frame ARP

En la Figura 28 en la sección Ethernet II se encuentra la información sobre la cabecera de la trama.

Destination: Broadcast (ff:ff:ff:ff:ff:ff): Indica que el destinario del paquete son todos los equipos de la red.

Address: Broadcast (ff:ff:ff:ff:ff:ff): Se indica que la dirección de destino es la dirección de máquina ff:ff:ff:ff:ff:ff que son todos los equipos de la red.

.... ..1. .... = LG bit: Locally administered address (this is NOT the factory default): Se indica que el bit LG está en 1 lo que significa que la configuración se modificó.

.... …1 .... = IG bit: Group address (multicast/broadcast): El bit IG indica si el usuario de destino o fuente es un solo equipo o todos los que conforman la red, y en este caso es el número 1 que significa que son todos los equipos de la red (broadcast) y si fuera un solo equipo (unicast) se coloca 0 al bit IG.

Source: 00:19:99:57:85:70 (00:19:99:57:85:70): Muestra la dirección MAC de destino.

Address: 00:19:99:57:85:70 (00:19:99:57:85:70): Dirección MAC destino.

.... ..0. .... = LG bit: Globally unique address (factory default): El bit LG es igual a 0 por lo cual indica se utiliza la configuración por defecto.

....0 .... = IG bit: Individual address (unicast): El bit IG es igual a 0 lo cual indica que es un solo equipo (unicast).

Type: ARP (0x0806): Indica el protocolo que se utilizó.

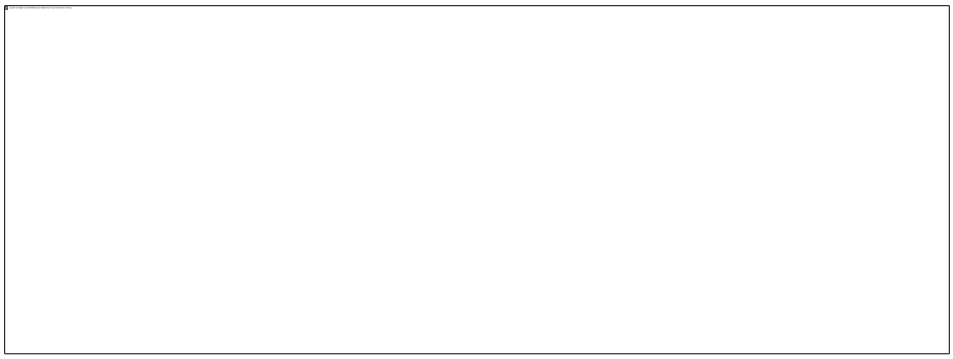


Figura 28 Detalle de la sección Ethernet ARP

En la figura 29 se muestra la sección del protocolo ARP que indica que es de tipo request (solicitud).

Hardware type: Ethernet (1): Indica que el tipo de Hardware que se utilizó es Ethernet.

Protocol type: IP (0x0800): Indica que se relacionará la dirección MAC a una dirección IPv4.

Hardware size: 6: Indica que la longitud de las direcciones MAC es de 6 bytes.

Protocol size: 4: Indica que el tamaño de la direcciones IPv4 es de 4 bytes.

Opcode: request (1): Indica que el paquete se trata de una petición o solicitud.

Sender MAC address: 00:19:99:57:85:70 (00:19:99:57:85:70): Indica la dirección física del equipo que envía la solicitud.

Sender IP address: 10.9.8.124 (10.9.8.124): Indica la dirección IP del equipo que hace la solicitud.

Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00): Indica que no se conoce la dirección física del destino por tal motivo se coloca 0.

Target IP address: 10.9.8.126(10.9.8.126): Muestra la dirección IP de la máquina de destino.

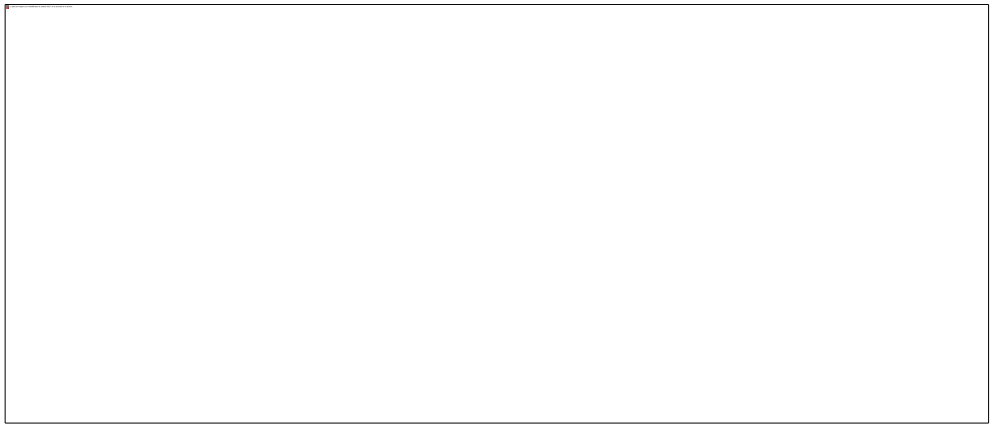


Figura 29 Detalle de la sección del Protocolo ARP

En el paquete 196 que es una respuesta a la petición del paquete 195, se comprueba que un host con IP 10.9.8.126 realiza una respuesta ARP donde indica cuál es su dirección MAC; la información de destino y el remitente se invierte, para mostrar que la respuesta ARP es ahora el remitente como se observa en la Figura 30.

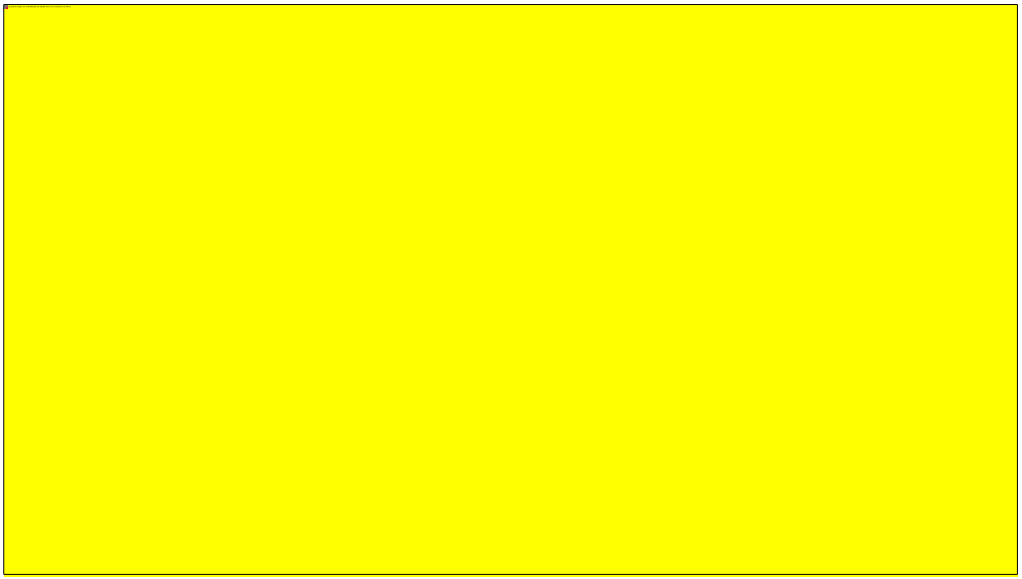


Figura 30 Captura Respuesta ARP

En la Figura 31 en la sección Frame muestra que el paquete 196 tiene un tamaño de 42 bytes.

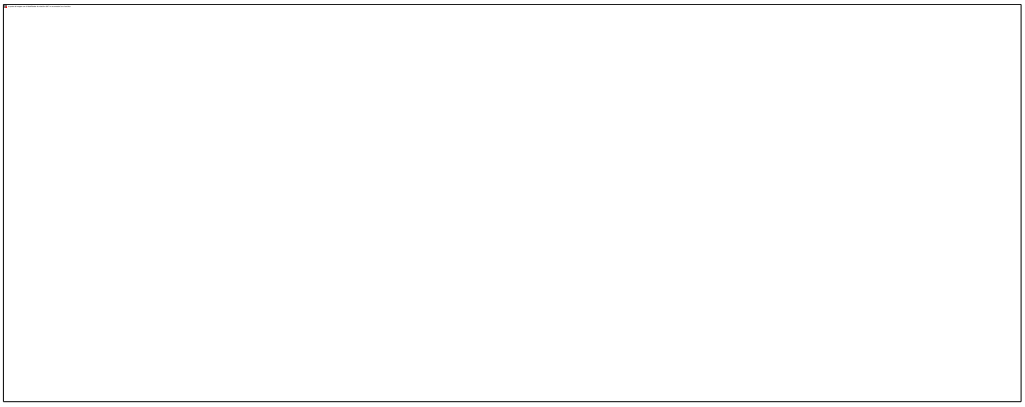


Figura 31 Detalle de la sección Frame respuesta ARP

En la segunda sección Ethernet II se distingue que el contenido de la respuesta ARP es diferente a una solicitud. Ver Figura 32.

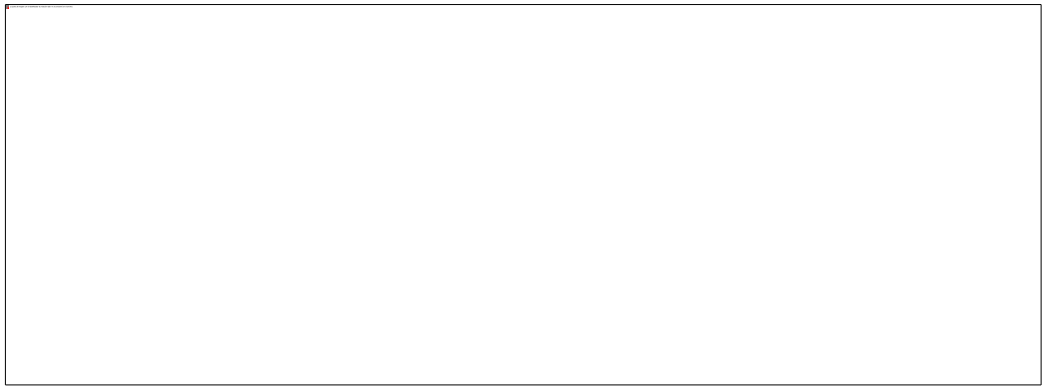


Figura 32 Detalle de la sección Ethernet II respuesta ARP

Destination: 00:19:99:57:85:70 (00:19:99:57:85:70): Indica la dirección MAC del destinatario del paquete.

Address: 00:19:99:57:85:70 (00:19:99:57:85:70): Indica la dirección MAC de la máquina de destino.

.... ..0. .... = LG bit: Globally unique address (factory default): El bit LG es igual a 0 por lo que se utilizó la configuración por defecto.

....0 .... = IG bit: Individual address (unicast): El bit IG es igual a 0 lo que indica que es un solo equipo (unicast).

Source: e8:9a:8f:1a:1e:93 (e8:9a:8f:1a:1e:93): Indicala dirección MAC de origen.

Address: e8:9a:8f:1a:1e:93 (e8:9a:8f:1a:1e:93): La dirección MAC de la fuente.

.... ..0. .... = LG bit: Globally unique address (factory default): El bit LG es igual a 0 por lo que se utilizó la configuración por defecto.

....0 .... = IG bit: Individual address (unicast): El bit IG es igual a 0 lo que indica que es un solo equipo (unicast).

Type: ARP (0x0806): Protocolo que se utilizó.

La sección del protocolo ARP como se detalla en la Figura 33, se evidencia que es un paquete reply (respuesta) y se muestra la información de la cabecera que es igual a la del paquete request (solicitud) con las siguientes diferencias:

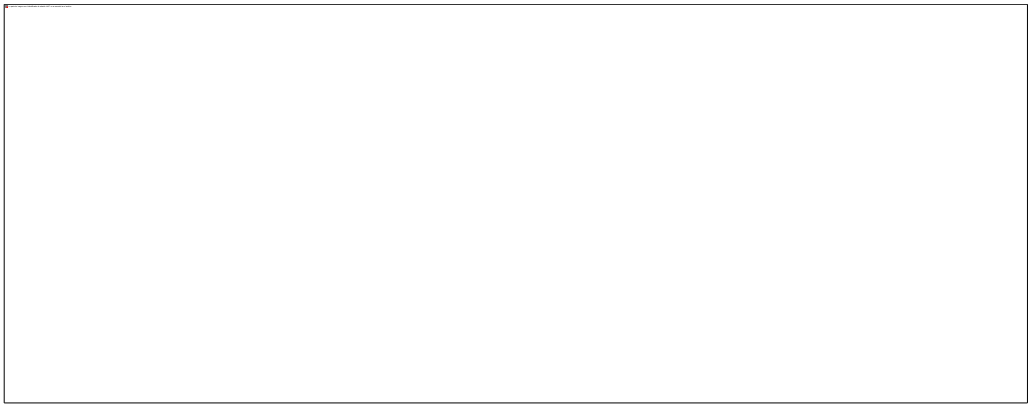


Figura 33 Detalle de la respuesta ARP

Opcode: reply (2): Indica que es una respuesta arp.

Sender MAC address: e8:9a:8f:1a:1e:93 (e8:9a:8f:1a:1e:93): Dirección MAC de la máquina que envía la respuesta arp.

Sender IP address: 10.9.8.126 (10.9.8.126): Dirección IP de la máquina que realiza la petición.

Target MAC address: 00:19:99:57:85:70 (00:19:99:57:85:70): Dirección MAC de la máquina destino.

Target IP address: 10.9.8.124 (10.9.8.124): Dirección IP de la máquina destino.

3.2.3 Envenenamiento ARP

El envenenamiento ARP (Protocolo de resolución de direcciones) es una técnica usada por atacantes en redes internas cuyo fin es obtener el tráfico de red circundante, aunque no esté destinado al sistema del propio intruso. Con este método, el atacante puede conseguir derivar la información hacia su propia tarjeta de red y así conseguir información sensible, bloquearla o incluso modificarla y mostrar datos erróneos a las víctimas.

Esta técnica no se basa en una vulnerabilidad concreta que pueda llegar a desaparecer con el tiempo, sino que se basa en un fallo de diseño de las redes TCP (Transmission Control Protocol), y por tanto, es un método de ataque siempre válido y eficaz a menos que se tomen medidas específicas contra él.

3.2.3.1 Condiciones del envenenamiento ARP

- La máquina atacante, conociendo las direcciones IP de los dos nodos cuyas comunicaciones se quieren intervenir, resuelve mediante ARP, si es necesario, las direcciones MAC que les corresponden.
- Mediante respuestas ARP, el atacante modifica el contenido de las cachés de las víctimas de forma que para la dirección IP de su interlocutor se corresponda la dirección MAC real del atacante.
- Cada vez que alguno de los nodos quiera enviar información al otro, resolverá la dirección MAC del mismo mediante su caché de ARP previamente envenenada, enviando así el tráfico al atacante en vez de al destinatario real.
- El switch enviará las tramas por la boca del destinatario, que en este caso es el atacante. Éste las recibirá y las pasará a la aplicación adecuada, que puede ser un

sniffer que capture todo el tráfico. Al estar todas las tramas destinadas a su dirección MAC, no es necesario que la tarjeta de red se encuentre en modo promiscuo.

- El atacante reenviará el contenido de las tramas al destinatario real. La única diferencia entre la trama original y la modificada es en un principio, la dirección ethernet del destinatario.
- El nodo correspondiente recibirá el tráfico como si nada hubiese ocurrido. El atacante, haciendo uso del envenenamiento ARP y la técnica del hombre en el medio o man in themiddle ha interceptado el tráfico sin que ninguno de los interlocutores se percate. (Pedraza, 2013)

3.2.3.2 Cómo funciona el envenenamiento ARP

El envenenamiento ARP se ejecuta en el transcurso de transacciones ARP, creando una condición de carrera, pero el envenenamiento más común se da con la distribución de respuestas ARP no solicitadas, que son almacenadas por los nodos en sus cachés ARP, generando de esta manera el escenario de cachés ARP envenenadas ver Figura 34. (Calle Espinoza, 2014)

Figura 34. Envenenamiento ARP

Fuente: (Calle Espinoza, 2014)

3.2.3.3 Métodos de envenenamiento a la caché ARP

La implementación del protocolo ARP es sencilla y tiene ciertas carencias que facilitan el uso ilegítimo del mismo. Las siguientes características son claves para que los métodos de envenenamiento a la caché ARP se lleven a cabo.

- **Protocolo sin estado.** Cuando una respuesta ARP es recibida por una máquina en la red local, ésta actualiza su caché ARP a pesar de no haber enviado una consulta ARP anteriormente.
- **Ausencia absoluta de autenticación en el protocolo.** Un computador modificará su comportamiento acorde con las tramas ARP recibidas, sin poder determinar de ningún modo la autenticidad de las mismas.
- **Cachés sujetas a alteraciones externas.** Es posible modificar los contenidos de una caché ARP tan sólo con construir y enviar una consulta o respuesta adecuada. (Pedraza, 2013)

A continuación se describen los métodos de envenenamiento a la caché ARP:

Respuesta no solicitada

Una respuesta ARP falsificada podría ser enviada a cualquier nodo y con esta respuesta, el nodo actualizará su caché ARP.

Una respuesta ARP falsificada también podría ser difundida a todas las computadoras que forman parte de la red local, envenenando de esta forma a la caché ARP de todas las computadoras con un solo mensaje.

Consulta

Cuando un computador recibe una consulta ARP, la capa ARP del mismo actualizará su caché ARP con el mapeo de los campos IP fuente y MAC fuente de la trama de consulta ARP, aún si la consulta no fue para ese computador. Es así que un atacante, solo necesita enviar una consulta ARP falsificada para envenenar la caché ARP de todas las máquinas en la red de área local.

Respuesta a una consulta

Un nodo malicioso en la red local, al recibir una consulta ARP legítima, puede enviar una respuesta ARP falsificada. Podría haber una condición de carrera entre la respuesta falsa y la legítima para alcanzar al computador solicitante; la caché ARP será actualizada con la última respuesta ARP recibida.

Los ataques mencionados anteriormente son a menudo usados como parte de otros serios ataques: denegación del servicio, suplantación de identidad, ataque de hombre en el medio. (Torres, 2012)

3.2.3.4 Identificación de intrusos con Wireshark

Con la captura del tráfico de la red se puede identificar los distintos casos de envenenamiento ARP, para lo cual se simula un ataque en el laboratorio. En la siguiente Figura 35 se observa el diagrama de la simulación del ataque. El atacante enviará reiteradamente respuestas ARP falsas hacia la víctima, para envenenar su cache ARP, en este ejemplo el atacante desea capturar el tráfico dirigido al Gateway.

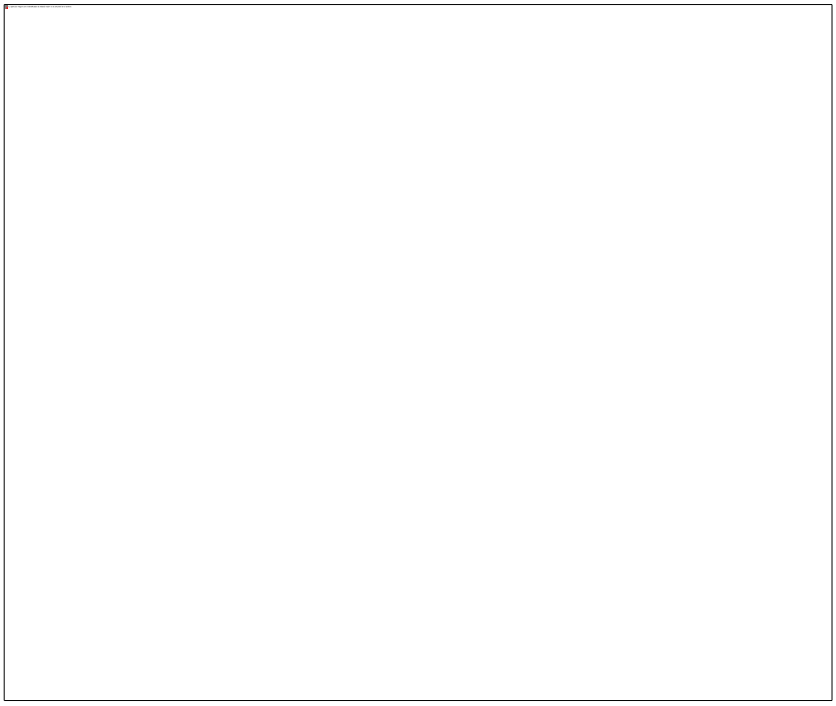


Figura 35 Diagrama de Red Ataque ARP

La información necesaria para realizar un ataque es contar con la IP de la víctima y la dirección MAC, la cual se puede obtener consultando la tabla de cache ARP con el comando ARP -a o mediante solicitudes ARP, a continuación se presenta en la Tabla 10 un resumen de los datos que se utilizaran en el ataque.

Tabla 10: Resumen de Datos para el Ataque

Equipo	**IP**	**Mac**	**IP hexadecimal**
Atacante	10.9.8.60	7C:05:07:FD:6A:43	0A 09 08 3C
Gateway	10.9.8.1	64:00:F1:E9:10:80	0A 09 08 01
Víctima	10.9.8.21	00:19:99:57:85:70	0A 09 08 15

Se comienza capturando el tráfico de la red en los laboratorios del DEPARTAMENTO DE CIENCIAS DE LA COMPUTACION, para seleccionar un

trama de una respuesta ARP, y se exporta dando clic derecho en Export Selected Packet Bytes para modificar la trama. Figura 36.

Figura 36 Captura Exportar Trama ARP

Una vez que se exporta el archivo, se modifica en un editor hexadecimal de la siguiente manera. Figura 37.

Figura 37 Trama ARP

Con la ayuda de Wireshark se identifica la posición de las direcciones MAC e IP a modificar para lo cual en el detalle de la respuesta ARP, se extiende la pestaña Ethernet y se selecciona Destination, marcándose la dirección MAC del destino dentro del archivo hexadecimal, que en este caso se sustituye por la dirección de la víctima. 00:19:99:57:85:70 como se observa en la Figura 38.

Figura 38 Sustitución MAC de la Víctima

Se procede a dar clic en la pestaña Source que indica la dirección Mac del remitente, y se suplanta por la del atacante 7C:05:07:FD:6A:43 como se muestra en la Figura 39.

Figura 39 Sustitución de la Dirección MAC del Remitente

Una vez terminado se extiende la pestaña de Address Resolution Protocol (Reply) y se seleccionaSender MAC Address, que significa la Mac del remitente y de igual forma que el caso anterior se reemplaza por la del atacante 7C:05:07:FD:6A:43. Figura 40.

Figura 40 Reemplazo Dirección MAC del atacante

En la opción Sender IP Address se sustituye por la dirección ip en hexadecimal del Gateway 0A 09 08 01. Figura 41.

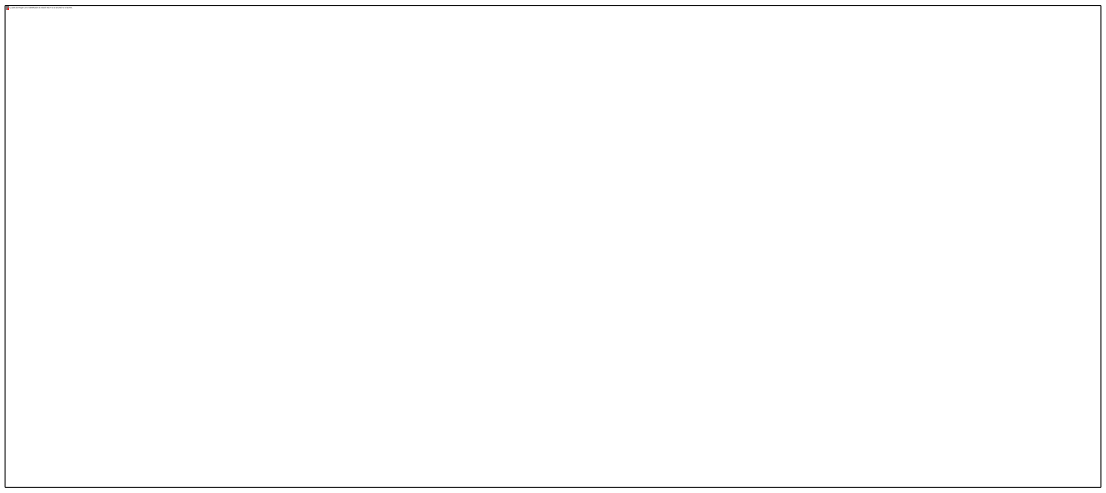


Figura 41 Sustitución por la Dirección del Gateway

En la opción Target MAC Address se sustituye por la dirección MAC de la víctima 00:19:99:57:85:70, como se muestra en la Figura 42.

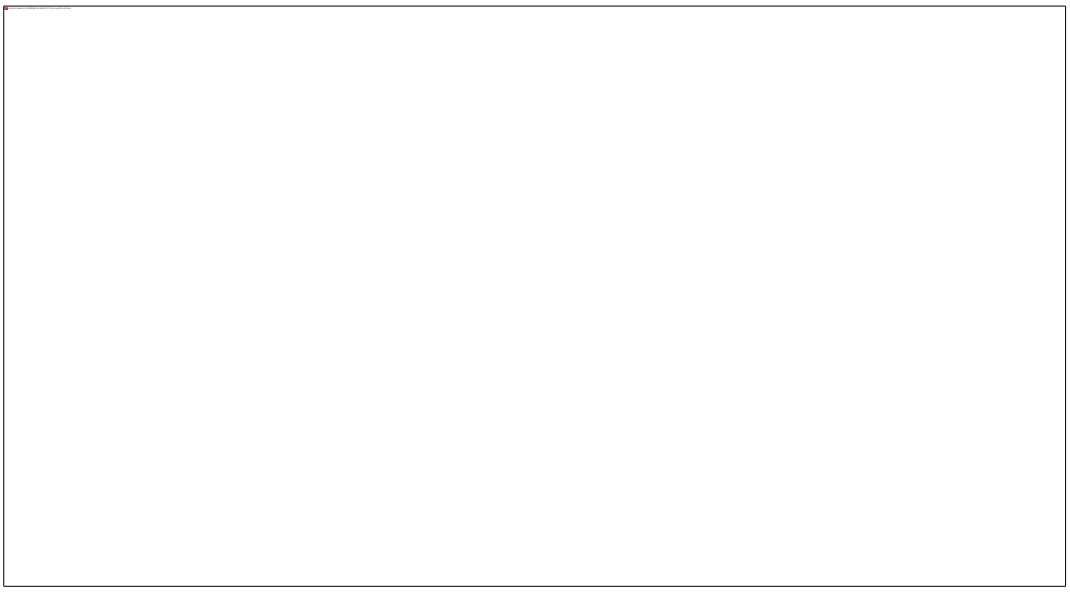


Figura 42 Sustitución por la MAC de la Vctima

La Figura 43 muestras la opción Target IP Address se coloca la IP en hexadecimal de la víctima y se guarda el documento.

Figura 43 IP Hexadecimal de la Víctima

Una vez que se cuenta con el archivo hexadecimal, mediante File2Cable en Linux se envía hacia la víctima para envenenar su cache ARP, con el comando sudo file2cable -v -i eth0 -f envenenamiento_ARP_Modificado, siendo eth0 el puerto por donde se envía el paquete y envenenamiento_ARP_Modificado el nombre del archivo como se observa en la Figura 44.

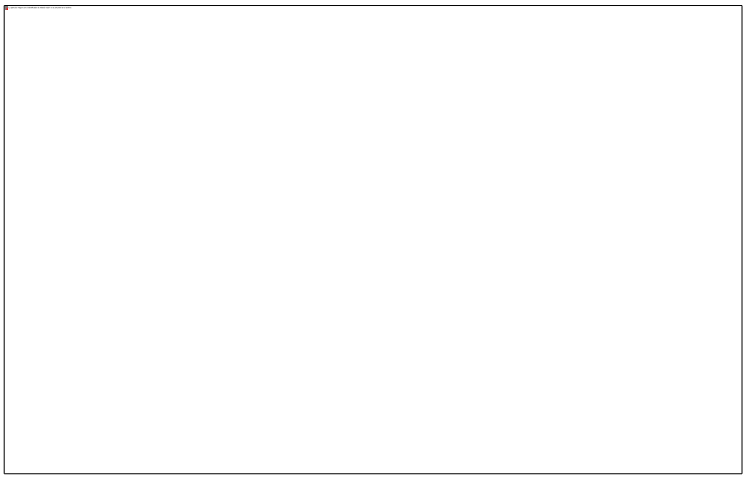


Figura 44 Envenenamiento cache ARP

Para comprobar si existió el envenenamiento se realiza una consulta al cache ARP de la víctima antes y después del ataque, además de capturar las tramas con Wireshark. Como se observa en la Figura 45, la MAC del Gateway es 64-00-F1-E9-10-80 antes del ataque.

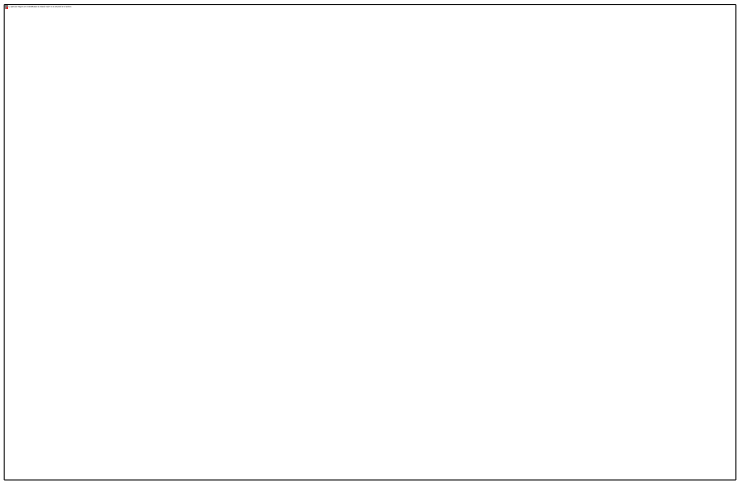


Figura 45 Consulta y Comprobación del Ataque

Luego de realizar el ataque la MAC del Gateway ha sido modificada por la del atacante 7C-05-07-FD-6A-43. Figura 46.

Wireshark es una herramienta que permite identificar el tráfico que circula por el puerto de monitoreo, obteniendo un reporte continuo de una posible “duplicación” de identidad, alertando al administrador para aislar el equipo afectado.

Es de suma importancia conocer las direcciones IP y las correspondientes MACs asociadas para cada máquina para identificar fácilmente la duplicación de identidad.

Figura 46 Verificación envenenamiento ARP

En la zona inferior de la Figura 46 se observa resaltado de color amarillo muestran que las direcciones IP se encuentran ocupando dos roles a la vez, que no le pertenecen al usuario y Gateway.

4. CAPÍTULO 4

4.1 Protocolo ICMP

Es un protocolo que permite administrar información relacionada con errores de los equipos en red. ICMP corrige los errores sino que los notifica a los protocolos de capas cercanas. Por lo tanto, el protocolo ICMP es usado por todos los routers para indicar un error (llamado un problema de entrega). (Kioskea, 2015)

4.1.1 Analizar tráfico normal del Protocolo ICMP

Se puede construir filtros de visualización que comparan valores usando un número diferente de operadores de comparación.

Los principales filtros utilizados en el protocolo ICMP se muestran en la Tabla 11:

Tabla 11 Filtros del protocolo ICMP

Filtro	Descripción
Icmp.lenght	Tamaño original del datagrama
Icmp.redir_gw	Dirección del gateway
Icmp.type	Tipo
Icmp.mip.seq	Número de Secuencia
Icmp.mpls.lenght	Tamaño
Icmp.mpls.ttl	Tiempo de vida
Icmp.mpls.version	Versión

El protocolo ICMP tiene aplicaciones mediante el proceso ping.

El proceso ping comprueba el estado de la conexión con uno o varios equipos para determinar si un sistema IP específico es accesible en una red. Es útil para diagnosticar los errores en redes.

La orden **ping** genera paquetes ICMP de tipo *echo request* y *echo reply*.

4.1.2 Proceso PING

En el laboratorio de multimedia del DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN, las computadoras están en red, se utilizó dos máquinas para capturar el tráfico y poder analizar el protocolo ICMP, la PC1 con IP 10.9.8.184 y la PC2 con IP 10.9.8.183, lo primero es iniciar Wireshark para comenzar a capturar el tráfico y después ingresar a la consola MS-DOS y hacer ping para ver si existe comunicación entre los dos equipos como en la Figura 47.

Figura 47 Verificar la conexión entre PCs

Una vez que se realizó el ping con la otra PC, se filtra en Wireshark por el protocolo ICMP para ver los mensajes de los paquetes tanto de envió como los de respuesta, ver Figura 48.

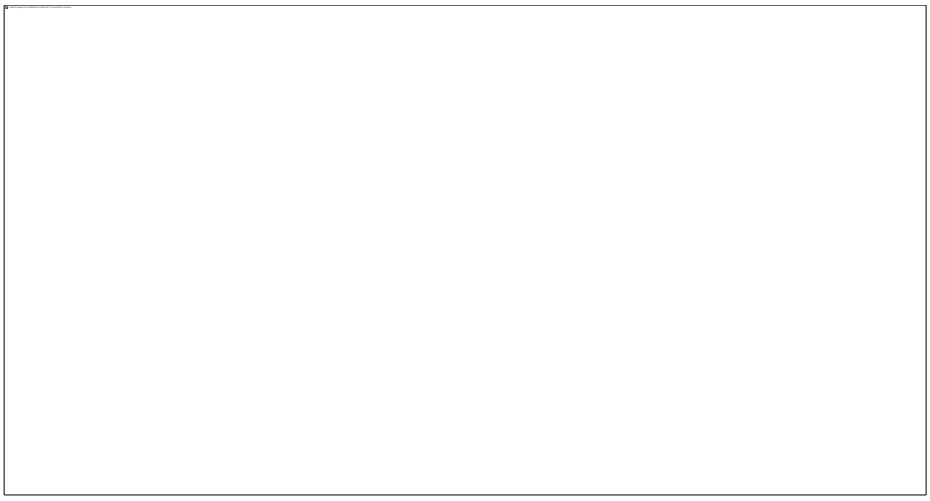


Figura 48 Wireshark Filtrado de Paquetes

La sección del Frame muestra el paquete 20 y su tamaño es de 74 bytes, la fecha que se realizó la captura del paquete, también se puede ver todos los campos con su respectiva información como se explicó en el protocolo TCP. Figura 49.

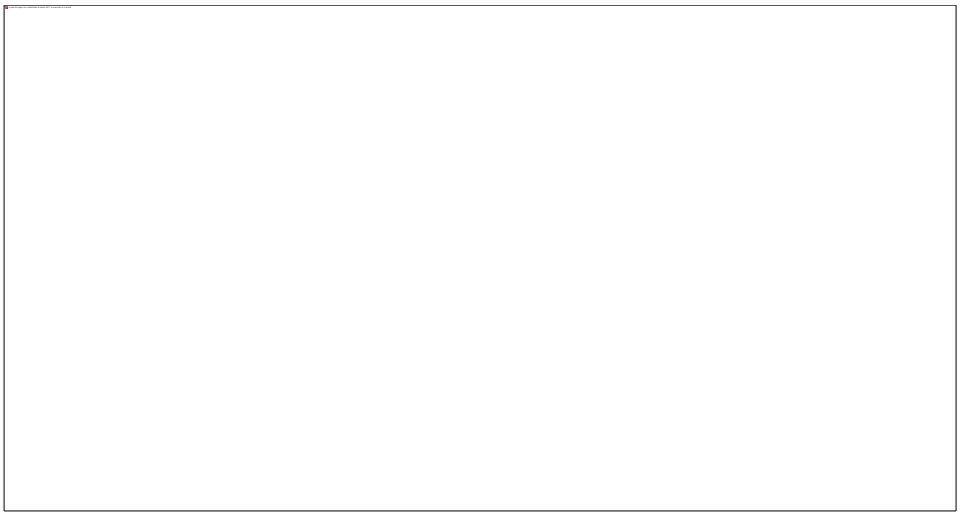


Figura 49 Detalle de la sección del Frame ICMP

En esta Figura 50 se nota la sección Ethernet II donde se puede ver que se visualiza una descripción de la dirección MAC de origen y destino, el contenido es igual a los protocolos mencionados en el capítulo anterior.

Figura 50 Detalle de la sección Ethernet ICMP

La sección siguiente es el protocolo IP y sus campos ya se explicaron en el capítulo anterior con la diferencia que tienen distinta información como por ejemplo el campo Protocol es igual a ICMP ver en la Figura 51.

Figura 51 Detalle de la sección IP ICMP

La Figura 52 muestra la sección del protocolo ICMP con sus respectivos campos que se detallan a continuación.

Type: 8 (Echo (ping) request): Indica el tipo de mensaje que se está haciendo, en este caso es de tipo 8 que significa un "echo request", si el tipo fuese 0 indica que es un "echo reply.

Code: 0: Indica que el subtipo de mensaje es 0.

Checksum: 0x4d32 [correct]: Indica que no hay errores dentro de los datos.

Identifier: identificador del mensaje.

Sequence Number: Número de secuencia que le corresponde al paquete.

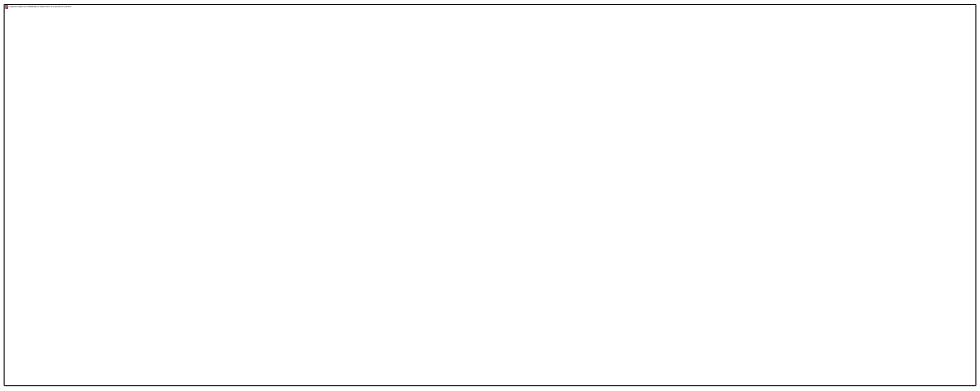


Figura 52 Detalle de la sección ICMP

4.1.3 Ataque ICMP Flooding

Es posible reducir el ancho de banda disponible de un host utilizando lo que se conoce como ICMP flooding (Inundación de mensajes ICMP). ICMP es el protocolo de mensajes de control de internet y se utiliza normalmente para verificar el estado de la red. Ver Figura 53.

La Figura 53 es el esquema del ataque, el computador atacante con Sistema Operativo Linux con IP 10.9.8.238 envia un brodcast a toda la red 10.9.8.0/24 para que las computadoras de la red le contesten con un mensaje ICMP a la víctima con IP 10.9.8.89.

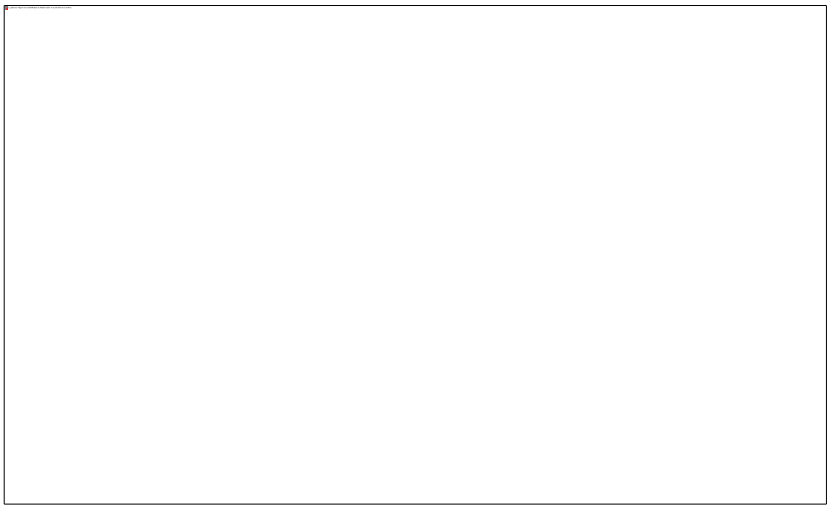


Figura 53 Diagrama de Red Ataque ICMP Flooding

Se envía un ataque desde el terminal de una PC con Linux como se observa en la Figura 54, la cual inunda el ancho de banda de la víctima con mensajes ICMP, colocando como destino el broadcast de la red y así generar una tormenta de broadcast.

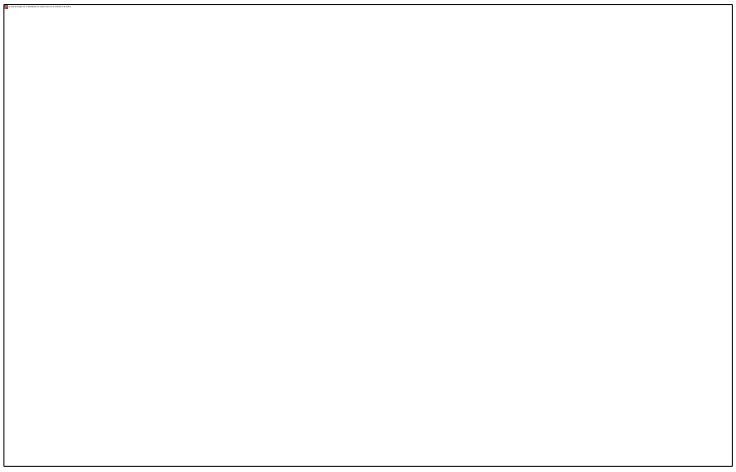


Figura 54 Ataque Comando Hping3

Como se observa en la Figura 55, se realizó un ataque ICMP donde solo respondieron los computadores Apple con direcciones IP 10.9.8.215, 10.9.8.218 y el Gateway, ya que el Sistema Operativo Windows está configurado para que automáticamente descarte los paquetes ICMP generados por hping3.

Estos son reconocidos por su encabezado. Se comprueba que se realizó el ataque ya que no se tuvo conexión a Internet. También se puede observar en el campo Info hay el mensaje Destination Unreachable esto quiere decir que el router considera la dirección IP destino como inalcanzable o el puerto especificado no está activo.

Figura 55 Ataque ICMP Flooding

En la Figura 56 muestra que la primera sección del Frame contiene el número del Frame 14, la longitud 126 bytes, nos marca con otro color el error es decir el paquete rechazado y el resto de los parámetros de la sección son iguales al contenido de los protocolos antes mencionados.

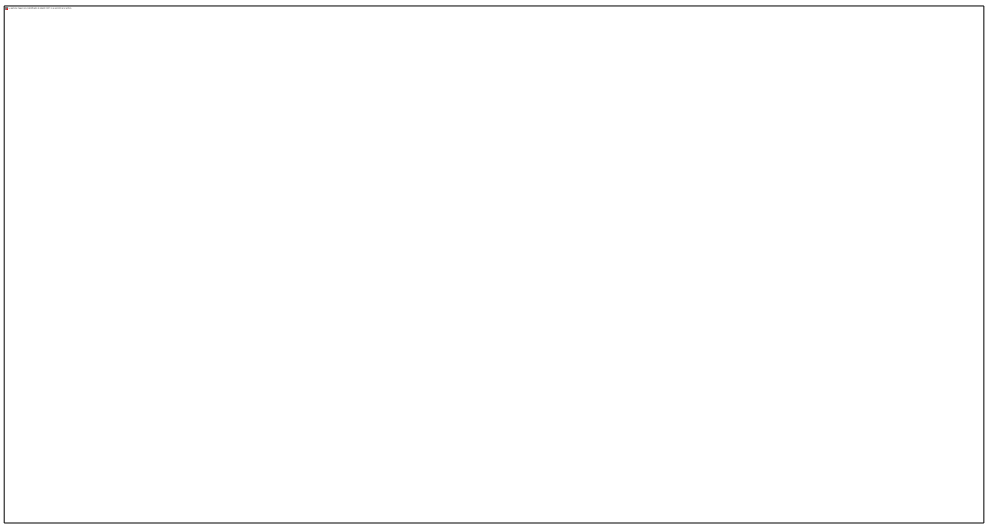


Figura 56 Descripción del frame ICMP

La Figura 57 muestra la sección del protocolo ICMP con sus respectivos campos que se detallan a continuación.

Type: 0 (Echo (ping) reply): Indica que el tipo de mensaje está haciendo un "echo reply" ya que es 0.

Code: 0: Esto quiere decir que la cabecera de ICMP es inválida.

Checksum: 0x7447: Indica que existe un error dentro de los datos ya que si estuviera todo correcto nos saldría un mensaje [correct].

Identifier: identificador del mensaje.

Sequence Number: Número de secuencia que le corresponde al paquete.
Los campos: identificador y el número de secuencia vuelven al inicio sin alterar.

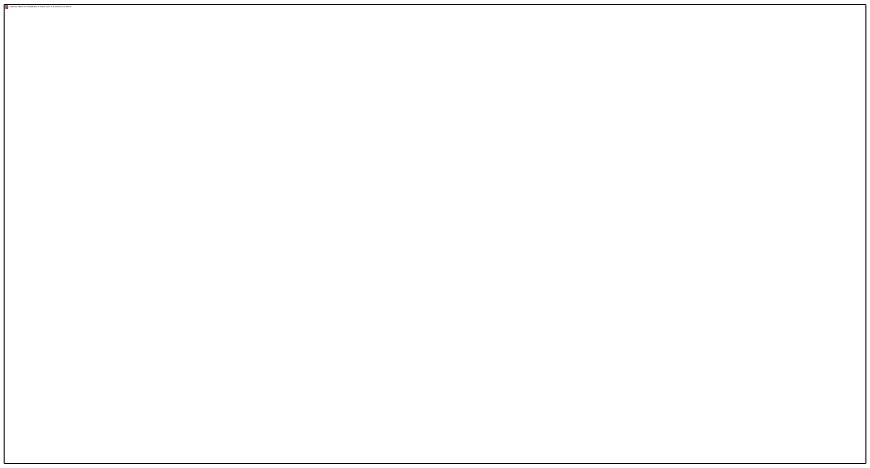


Figura 57 Descripción del detalle del paquete

En la Figura 58, es evidente que en el momento que se realiza el ataque el ancho de banda consumido aumenta considerablemente dejando así a toda la red sin acceso a Internet.

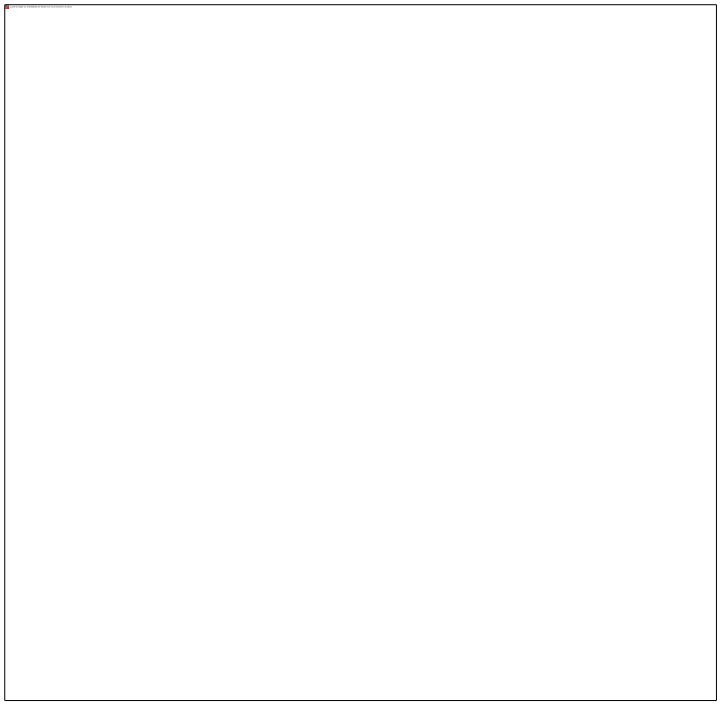


Figura 58 Estado del Rendimiento de la Red

4.2 Protocolo UDP

UDP es un protocolo no orientado a conexión. Es decir cuando una maquina A envía paquetes a una maquina B, el flujo es unidireccional. La transferencia de datos es realizada sin haber realizado previamente una conexión con la máquina de destino (maquina B), y el destinatario recibirá los datos sin enviar una confirmación al emisor (la maquina A). (Yazid, 2012)

4.2.1 Analizar tráfico normal del Protocolo UDP

El protocolo UDP es útil en situaciones cliente-servidor, a menudo el cliente envía una solicitud al servidor y espera una respuesta. Si se pierde la solicitud o la respuesta, el cliente puede terminar y probar de nuevo. El código es simple y se necesitan pocos mensajes en comparación con otros protocolos. Algunos Protocolos que usan a UDP son: TFTP, SNMP, DHCP y DNS.

Se puede construir filtros de visualización que comparan valores usando un número de diferentes operadores de comparación como se muestra en la Tabla 12.
Entre los principales filtros utilizados en el protocolo UDP se tiene:

Tabla 12 Filtros del protocolo UDP

Filtro	Descripción
Udp.dstport	Puerto de Destino
Udp.port	Puerto de Origen y Destino
Udp.srcport	Puerto de Origen
Udp.lenght	Tamaño
Udp.proc.dstcmd	Nombre del proceso de destino
Udp.proc.dstuname	Nombre de usuario del proceso de destino
Udp.procsrcuname	Nombre de usuario de proceso de origen

Para iniciar el análisis del protocolo UDP, se captura el tráfico en el laboratorio para poder examinar el segmento UDP mediante una consulta DNS y observar los paquetes que se generaron al comunicarse con un servidor. La figura 59 muestra el paquete 234, el cual indica que la IP 10.9.8.236 realizó una consulta estándar al servidor DNS para ingresar a la URLwww.hotmail.com.

En el paquete 252 el servidor DNS responde la consulta estándar a la máquina que la realizo.

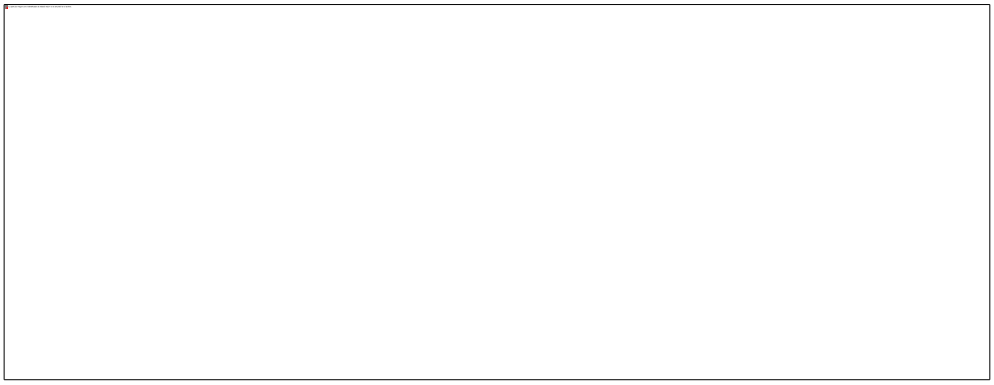


Figura 59 Captura tráfico UDP

En la siguiente Figura 60 se describe la primera capa Frame del panel de detalle, la cual dice que el paquete es el 234 y su tamaño es de 75 bytes como se muestra en la línea seleccionada, esta es la cantidad de bytes que se usa para enviar la consulta DNS al servidor de nombres que solicita la dirección IP de www.hotmail.com, los demás campos de esta sección ya se encuentran explicados en el capítulo 1 lo que varía es la información.

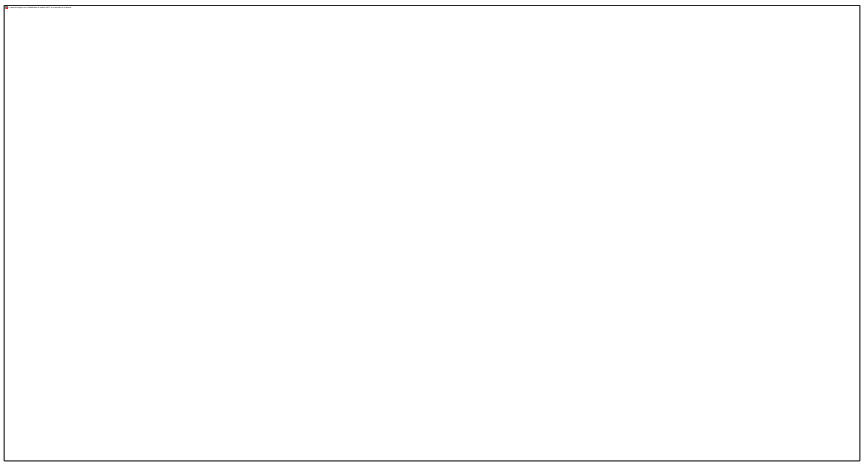


Figura 60 Detalle de la sección Frame DNS

En la segunda capa Ethernet II se muestra la dirección MAC de origen la cual proviene de la PC local en donde se origina la consulta DNS y la dirección MAC del destino que en este caso es el servidor DNS. Figura 61.

Figura 61 Detalle de la sección Ethernet II DNS

La sección del protocolo IP como se ve en la Figura 62 muestra la dirección IP de origen de la consulta DNS 10.9.8.236 y la dirección IP del destino 10.1.0.104. En este caso la IP de destino es el gateway predeterminado en esta red.

Figura 62 Detalle de la sección IP DNS

En la sección del protocolo UDP como se puede observar en la Figura 63 consta de 4 campos que son el encabezado del paquete, cada campo del encabezado consta de 16 bits.

Source Port: 58724 (58724): Indica el puerto de origen aleatorio que la PC género el cual es el número de puerto que no están reservados.

Destination Port: 53 (53): Indica el puerto de destino en este caso es el puerto reservado del servidor DNS. En este puerto los servidores DNS escuchan consultas DNS de los clientes.

Length: 41: Indica la longitud del paquete UDP, que en este caso es 41, de los cuales 8 se utilizan para el encabezado y los 33 se utilizan como datos de consulta DNS.

Checksum: 0x1d98: Indica la integridad del paquete UDP.

UDP no tiene campos asociados con el protocolo de enlace de tres vías eso indica que el encabezado UDP tiene una sobrecarga baja, y cualquier problema que ocurra con la confiabilidad de transparencia de datos se debe solucionar en la capa de aplicación.

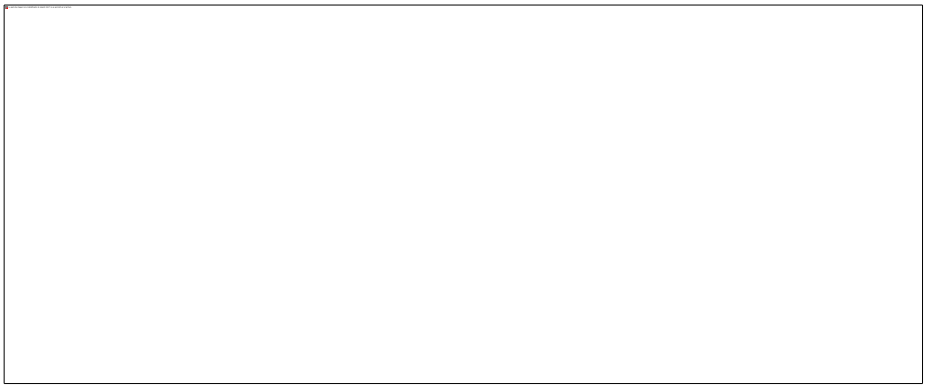


Figura 63 Detalle de la sección UDP DNS

En la última sección que representa al protocolo DNS se observa las banderas, la PC envio una consulta DNS para traducir la url introducida en una direccion IP y recibio

una respuesta DNS del paquete 252 para informar la direccion IP de www.hotmail.com. Figura 64.

Figura 64 Detalle de la sección DNS

4.3 Protocolo IP

El protocolo IP es el encargado de hacer llegar a su destino cada una de los paquetes, él memoriza de dónde vienen y cuál es su periodo de caducidad. El trabajo conjunto de los dos protocolos hace que la información llegue a nuestro ordenador desde cualquier parte del mundo y en muy poco tiempo.

El Protocolo IP no provee ningún mecanismo para determinar si un paquete alcanza o no su destino y únicamente proporciona seguridad (mediante checksums o sumas de comprobación) de sus cabeceras y no de los datos transmitidos. Por ejemplo, al no garantizar nada sobre la recepción del paquete, éste podría llegar dañado, en otro orden con respecto a otros paquetes, duplicado o simplemente no llegar. Si se necesita fiabilidad, ésta es proporcionada por los protocolos de la capa de transporte, como TCP.

Se puede construir filtros de visualización que comparan valores usando un número de diferentes operadores de comparación.
Los principales filtros utilizados en el protocolo como se muestra en la Tabla 13:

Tabla 13 Filtros del Protocolo IP

Filtro	Descripción
Ip.id	Identificación
Ip.len	Tamaño Total
Ip.opt.addr	Dirección IP
Ip.opt.len	Tamaño
Ip.src_rt	Ruta Fuente
Ip.src_host	Host Fuente
Ip.ttl	Tiempo de Vida

4.4 Suplantación de una Pagina Web

La siguiente práctica consiste en direccionar los usuarios hacia un clon de una página especifica la cual será alojada en el computador atacante direccionando así a todos los usuarios que quieran ingresar a la misma, donde el atacante podrá capturar todos los datos ingresados por las víctimas, como son credenciales de accesos.

La Figura 64 es el esquema del ataque, donde el computador atacante con Sistema Operativo Linux tiene la IP 10.9.8.238 clonará el Sitio Web de mi ESPE, además de realizar un ataque DNS Spoofing, siendo las víctimas todos los usuarios de la red 10.9.8.0/24.

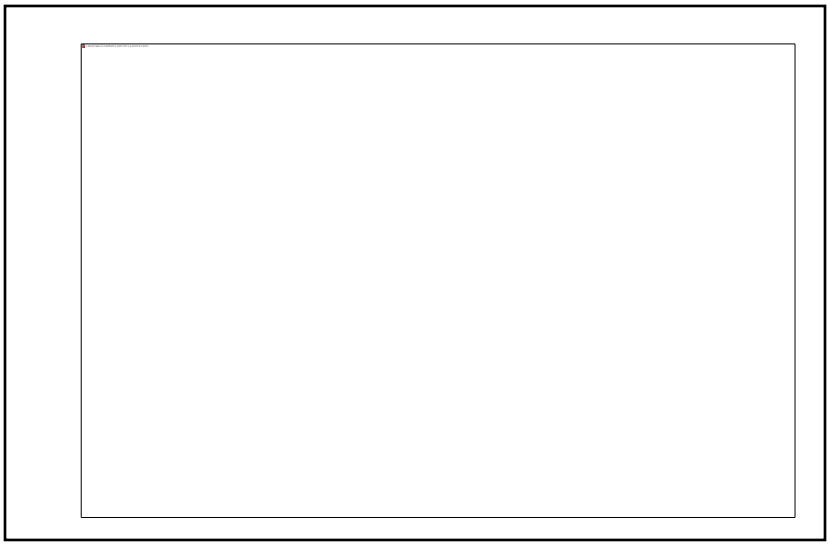


Figura 65 Suplantación de una Página Web

4.4.1.1 Clonación de un Sitio Web

1. Ingresar al terminal, ejecutar el comando sudo Setoolkid mediante la ejecución del mismo se obtendrá lo siguiente ver Figura 66.

Figura 66 Comando Sudo Setoolkid

2. En el menú que se generó se elige la primera opción la cual se basa en los ataques de ingeniería social (obtener información de los usuarios a través de manipulación), se despliega el siguiente menú ver Figura 67.

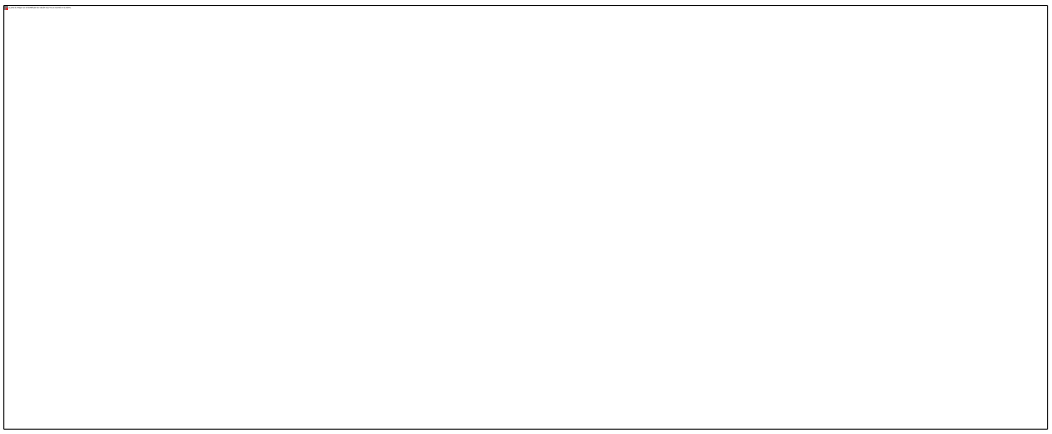


Figura 67 Menú Ataque Ingeniería Social

3. En el menú que se despliega se escoge la opción 2, se basa en un ataque web el cual se permitirá generar automáticamente un sitio falso con el cual se puede engañar a los destinatarios como se puede observar en la Figura 68.

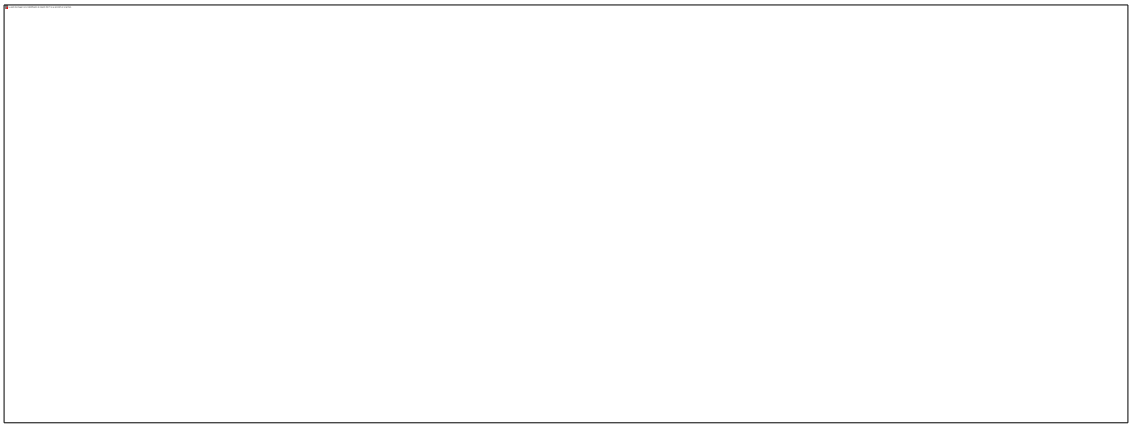


Figura 68 Menú Ataque Web

En la pantalla siguiente ver Figura 69 se elige la tercera opción la misma que se basa en el Método Credencial Ataque Harvester (Ataque social para suplantar páginas web).

Figura 69 Menú Ataque Harvester

En la siguiente pantalla ver Figura 70, se elige del menú la opción Clonar Sitio (permite realizar ataques a usuarios que ingresen a una dirección web especificada (por medio de ingeniería social)).

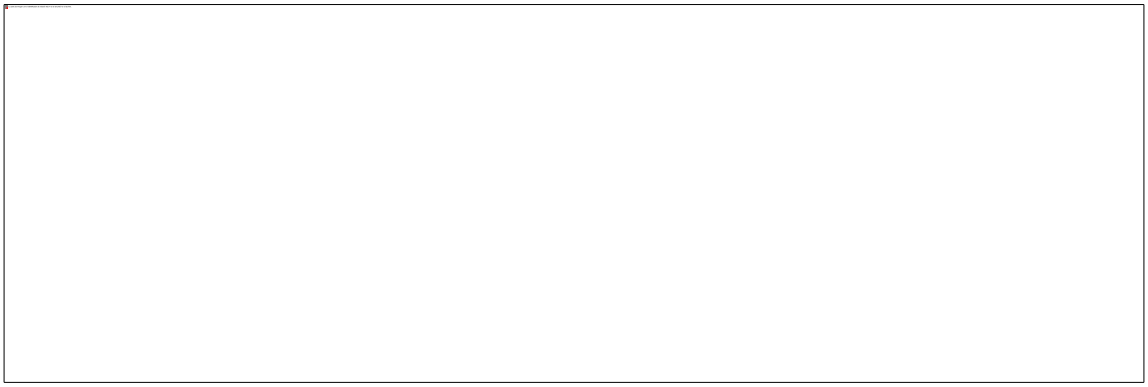


Figura 70 Menú Clonar Sitio

En la pantalla siguiente ver Figura 71, añadir la IP de la máquina desde la cual vamos a realizar la clonación de la página web que se desea obtener los datos, en este caso se utiliza la IP 10.9.8.238.

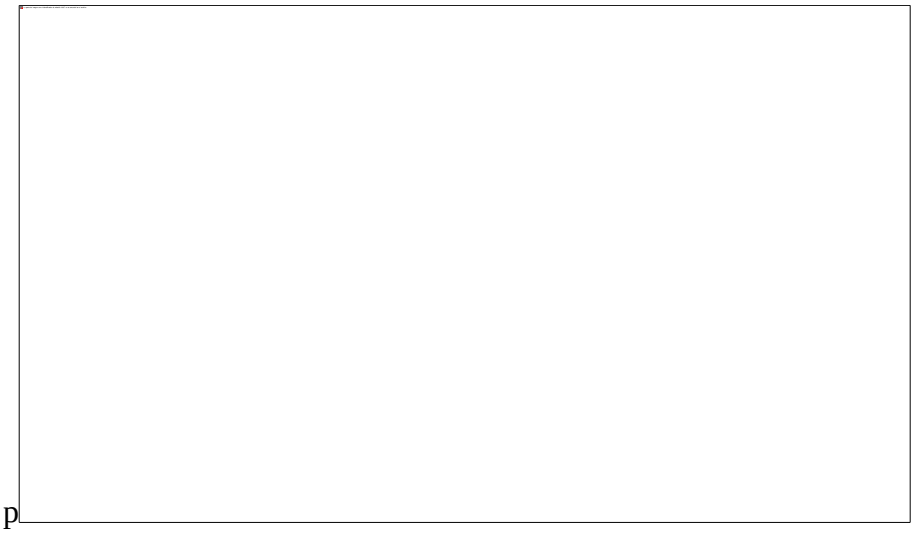
p 

Figura 71 Suplantación Página Web

Se añade el URL de la página de la cual se va a realizar la clonación en este caso se utiliza: https://miespe.espe.edu.ec/cp/home/displaylogin Ver en la Figura 72.

Figura 72 URL Pagina Clonación

Se muestra el proceso del clonado de la página web en el Figura 73, indica en que directorio se guarda la página creada, si el servidor apache está funcionando correctamente, al final se indica que todos los archivos han sido copiados y que la clonación de la página se realizó con éxito.

Figura 73 Clonación de Página Web

Para corroborar la clonación se escribe la dirección IP del computador atacante en una ventana del navegador web, en la cual debe salir la página clonada, en este caso el sitio web de Mi Espe. Ver Figura 74

Figura 74 Comprobación Sitio Web

4.4.1.2 DNS Spoofing

La principal tarea de un servidor DNS es convertir las direcciones IP en algo mucho más comprensible y viceversa, mejorando notablemente la experiencia en la red.

Un ejemplo de esto sería la propia web de la universidad, www.espe.edu.ec 192.88.58.167, accesible por ambas formas, pero la primera es mucho más fácil de recordar que la segunda. El servidor DNS es el encargado de "traducir" el nombre a la dirección numérica. Aprovechando esta necesidad se puede crear respuestas ARP falsas para dirigir a la víctima a la dirección IP donde se aloja el clon del Sitio Web.

La herramienta a usar para realizar la práctica se llama ettercap, es un interceptor/sniffer/registrador para LANs con switch. Primero se introduce el comando sudo gedit /etc/ettercap/etter.dns como consta en la Figura 75, para ingresar la IP del DNS falso.

Figura 75 Comando ettercap

Al ejecutar el comando gedit – ettercap se despliega el editor de texto, para ingresar la URL a suplantar y la dirección IP del servidor que aloja la web clonada como se indica en la Figura 76.

Figura 76 Valores DNS Spoofing

Se ejecuta el comando sudo ettercap –T –q –M arp –i eth0 –P dns_spoof // //, para comenzar el ataque. Ver Figura 77.

-T: Selecciona la interfaz de usuario solo texto.
-q: No muestra el contenido del paquete.
-M: Realiza un ataque MITM (Man in the middle).
-i: Interfaz de red.
-P: Carga los plugin.

Figura 77 Comando Ataque ettercap

Cada vez que la víctima ingrese a la URL seleccionada para la suplantación, la herramienta mostrara un mensaje indicando que se tuvo éxito al realizar el ataque DNS Spoofing indicando a que dirección IP fue redireccionada como se observa en la Figura 78.

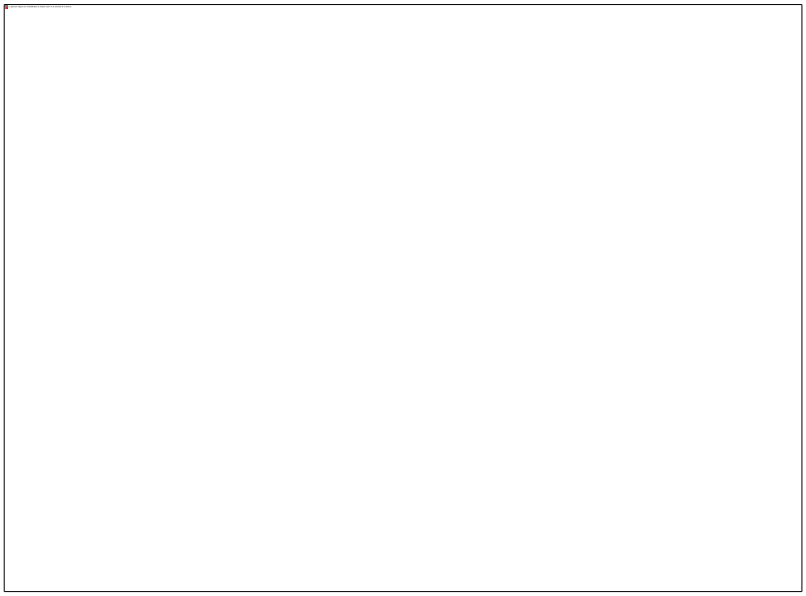


Figura 78 Ataque exitoso

La víctima cada vez que trate de ingresar a la página de google, será direccionada al servidor donde se encuentra alojada la página clon de Mi Espe. Ver Figura 79.

Figura 79 Resultado DNS Spoofing

4.4.1.3 Robo de Credenciales

Al estar realizando el ataque DNS Spoofing, es posible capturar el usuario y contraseña de cualquier víctima que intente loguearse, para lo cual se simula con un Usuario: pruebaUsuario y una Clave: pruebaClave.Ver Figura 80.

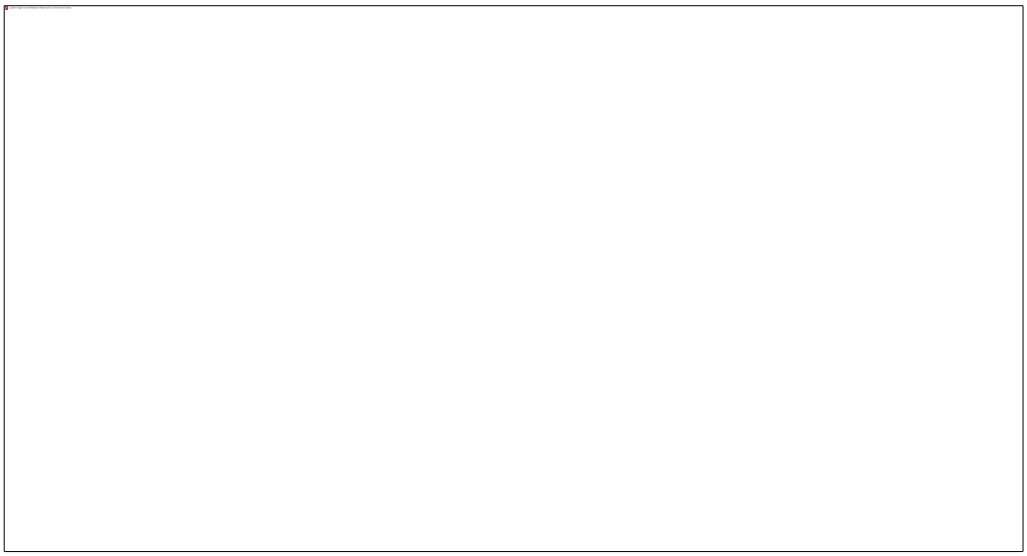


Figura 80 Ingreso de Credenciales

Al presionar en el botón ingresar la víctima enviará un paquete HTTP POST con el usuario y la contraseña hacia el atacante, y con la ayuda de Wireshark se puede capturar y visualizar este paquete. Ver Figura 81.

Figura 81 Captura Usuario y Contraseña

Como se muestra en la Figura 81, el paquete 434 contiene el usuario y contraseña antes ingresados de la víctima.

5. CAPÌTULO 5

5.1 CONCLUSIONES

Se realizó un análisis comparativo de las herramientas y se obtuvo como resultado que Wireshark es una herramienta adecuada, completa, confiable, compatible con distintos sistemas operativos y de gran facilidad de uso. El software implementado para el análisis de la red fue Wireshark el cual permitió identificar el equipo que origino los ataques y el protocolo que se utilizó, se puede generar filtros para reconocer los distintos tipos de ataques. A los administradores de la red les permite detectar, analizar o solucionar anomalías,

Del análisis realizado con la herramienta Open Source Wireshark, se determinó que existen vulnerabilidades al realizar diferentes tipos de ataques como son SYN Flooding el cual se lo realizó mediante el uso del Sistema Operativo Linux y el Comando Hping3 el cual genera y envía paquetes desde un origen falso manteniendo así en modo de espera a la víctima y dejándolo sin servicio de Internet.

Al simular el Envenenamiento ARP se determinó que los usuarios son vulnerables y pueden ser víctimas de ataques, como es la duplicación de identidad donde el atacante puede enviar reiteradas respuestas ARP hasta envenenar la cache de sus víctimas. En este ataque Wireshark permite identificar el tráfico que circula por el puerto de monitoreo, obteniendo así un reporte continuo de una posible duplicidad de identidad alertando al administrador para aislar el equipo afectado.

El ataque ICMP Flooding utiliza la herramienta ping, Sistema Operativo LINUX y el comando Hping3 el cual permite enviar un paquete hacia la dirección del broadcast con la IP a atacar, por lo cual toda la red responderá a la víctima disminuyendo así el ancho de banda, siendo vulnerables a este tipo de ataques los equipos con Sistema

Operativo IOS ya que Windows descarta automáticamente los paquetes ICMP que tienen un encabezado incorrecto

En el ataque DNS Spoofing, se realizó la suplantación de la página web de MI ESPE mediante el sistema operativo LINUX y la utilización del comando Ettercap el cual permitió enviar resoluciones falsas de DNS hacia la víctima para luego re-direccionarlo hacia una página clonada y mediante la técnica sniffing capturar las credenciales para obtener acceso a información sensible de los usuarios.

5.2 RECOMENDACIONES

Es necesario que las personas encargadas de los laboratorios utilicen una herramienta sniffing para poder minimizar posibles vulnerabilidades o ataques que puedan estar ocurriendo, con lo que permitirá salvaguardar la información de acceso a sitios sensibles como son instituciones financieras, etc.

Es preciso que la contraseña que se utilice para acceder a información sensible sea fuerte es decir tenga una longitud al menos de 8 caracteres y una complejidad, que posea dígitos, letras y caracteres especiales y también es recomendable cambiarla con cierta regularidad para evitar el robo de la misma.

Para mayor seguridad de los usuarios se recomienda que al ingresar a una página Web donde se necesite digitar su usuario y contraseña lo realice por medio de un buscador y no mediante links los cuales son alterados para así evitar acceder a una página clonada o maligna y que tenga como objetivo robar sus credenciales.

Es recomendable realizar el análisis del tráfico en los Laboratorios de Ciencias de la Computación continuamente para prevenir posibles problemas y evitar perdida, suplantación o manipulación de la información y así disminuir el tiempo de corrección de errores

Para mejorar la seguridad se recomienda tener una línea base del funcionamiento normal de la red, mediante esto se puede hacer comparaciones y poder descubrir algún error o irregularidad a tiempo y así poder actuar frente a esta amenaza.

6. Bibliografía

A.M, N. (2013). *Protocolo ARP*.

Barahora, E., & Gellibert, P. (2011). *Analizador de Tráfico de Red.* Guayaquil: Escuela Superior Politécnica del Litoral.

Brassfield, C. (2012). *AMENAZAS A LA SEGURIDAD INFORMATICA.* RIOBAMBA: UNIVERSIDAD INTERAMERICANA .

Calle Espinoza, S. (2014). *Envenenamiento ARP*. Santa Cruz de la Sierra - Bolivia: Universidad Tecnológica Privada de Santa Cruz.

Callejas, A. (29 de 03 de 2010). *Como funciona? TCP/IP*. Obtenido de http://www.rootzilopochtli.com/2010/03/como-funciona-tcp-ip/

Candelas, F. (2008). *Protocolos de Transporte TCP y UDP*. Alicante: Univeridad de Alicante.

Cerda, D. (2014). *Arquitectura TCP/IP*. Tijuana.

Chacon, E., Neto, A., & Vega, J. (Septiembre de 2011). *ESPE Comunicación de Datos 8vo Nivel*. Obtenido de ESPE Comunicación de Datos 8vo Nivel: http://comunicdatoschaconnetovega.blogspot.com/p/comparacion-entre-modelo-tcpip-y-osi.html

Chen, T. (2011). *El protocolo ARP. Protocolo de resolución de dirección.* Mexico: Apuntes de Networking ©.

Compostela, J. (11 de 04 de 2013). *CAPA FÍSICA (CAPA 1) DEL MODELO OSI*. Obtenido de http://www.ciclodeinformatica.es/2013/04/capa-fisica-capa-1-del-modelo-osi.html

Corletti Estrada, A. (2011). *Seguridad por Niveles.* Madrid: darFe.

De La Cruz, L. (2012). *Protocolo TCP/IP*. Tutoria Virtual de Javir Barragan Piña.

Franco, D., Perea, J. L., & Puello, P. (2011). *Metodología para la Detección de Vulnerabilidades en Redes de Datos.* Cartagena: Universidad de Cartagena, Facultad de Ingeniería, Grupo de Investigación en Tecnologías de las Comunicaciones e Informática, GIMATICA.

Garcia, A. (2012). *Protocolos de Interconexión de Redes.* Creative Commons 3.0 BY-NC-SA.

Garcia, I., & Casillas, J. C. (Febrero de 2015). *Informática Primitiva*. Obtenido de Informática Primitiva: http://informaticaprimitiva.blogspot.com/2015/02/udp-en-el-modelo-osi.html

Gil Vasquez, P. (2011). *Protocolo de Mensajes de Control de Internet (ICMP).* Alicante: Grupo de Innovacion Educativa en Automatica.

Graciano, A. (2014). *Protocolo IP.*

Jean, F. (Marzo de 2014). *Kioskea*. Obtenido de http://es.kioskea.net/contents/275-protocolos

Kioskea. (Febrero de 2015). *Kioskea*. Obtenido de Kioskea: http://es.kioskea.net/contents/265-el-protocolo-icmp

Llagua, A. (Domingo de Diciembre de 2012). *Alejandro Llagua*. Obtenido de Alejandro Llagua: http://alejollagua.blogspot.com/2012/12/el-modelo-tcpip.html

Monroy, E. (2013). *AMENAZAS INFORMATICAS.* MEXICO: ALFAOMEGA.

Moreno, J. (2008). *Guia de uso del Software Wireshark pra captura de tramas Ethernet.* Schneider.

Neto, A. (16 de 12 de 2011). *Networks On Chips - NOCs*. Obtenido de http://comunicdatoschaconnetovega.blogspot.com

Ordóñez, O. (2012). *Implementación de un prototipo mediante la utilización de un Software Libre para evitar ataques al protocolo ARP en una red de áreal local.* Riobamba.

Pedraza, L. (2013). *Envenenamiento ARP*.

Perez, B. (2011). *TIPOS DE INTRUSOS INFORMATICOS.* Caracas.

Perez, L. (Viernes de Octubre de 2012). *Liliana Perez*. Obtenido de Liliana Perez: http://liliperez160.blogspot.com/2012/10/modelo-tcpip.html

Pinzon, F. (1 de Septiembre de 2014). *slideshare.* Obtenido de slideshare: http://es.slideshare.net/fabianandradepinzon/unidad-1-introduccion-a-las-redes-de-computadores-38544212

Reyes, R. (2014). *FUNCIONALIDAD, DISEÑO, SIMULACION Y CONFIGURACION DE DISPOSITIVOS PARA UNA RED MPLS EN ENTORNO IPV6.* QUITO: UNIVERSIDAD POLITECNICA SALESIANA.

Rodriguez, J. F. (2012). *Modelo iso protocolos.*

Rodriguez, L. (26 de Septiembre de 2013). *Intalacion de redes*. Obtenido de Intalacion de redes: https://louisrodriguezh411.wordpress.com/2013/09/

Ruiz, J. M. (2015). *Analizadores de protocolos. Manual de Wireshark.* Madrid: Universidad de Alcalá.

Salamanca, H. (Jueves de Octubre de 2014). *Hosman Salamanca*. Obtenido de Hosman Salamanca: http://hosmansalamanca.blogspot.com/2014/10/protocolos-de-internettcpip.html

Sanchez, S. (2014). *IMPORTANCIA DE IMPLEMENTAR EL SGSI EN UNA EMPRESA CERTIFICADA BASC.* BOGOTA: UNIVERSIDAD MILITAR NUEVA GRANADA.

Seoane, M. (2011). *Redes de Computadoras.* Lima.

Systems, C. (2013). *Uso de Wireshark para examinar una captura de UDP y DNS.* Massachusetts: CISCO NETWORKING ACADEMY.

Teldat, R. (2010). *Protocolo ARP e InARP.*

Torres, J. (2012). *Suplantacion de identidad (ARP-DNS SPOOFING).*

Ubenga, J. J. (2011). *TIPOS DE ATAQUES Y VULNERABILIDADE EN UNA RED.* BADAJOZ.

Velazquez, K. (2014). *TCP (TRANSMISSION CONTROL PROTOCOL).* CARACAS.

Verdejo Alvarez, G. (2012). *SEGURIDAD EN REDES IP: Los protocolos TCP/IP.*

Vives, A. (2011). *WALC2011: DESPLIEGUE DE IPv6.* GUAYAQUIL: THE IPv6 COMPANY.

Yazid, H. (2012). *Protocolos de Transporte y Aplicación.*

Zeas Marín, R. C. (2011). *Analisis y Captura de paquetes de datos en una red mediante la herramienta WireShark.* Quito.