

ESCUELA POLITÉCNICA DEL EJÉRCITO

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

**CARRERA DE INGENIERÍA EN ELECTRÓNICA y
TELECOMUNICACIONES**

**PROYECTO DE GRADO PARA LA OBTENCIÓN
DEL TÍTULO EN INGENIERÍA**

**ANÁLISIS DEL COMPORTAMIENTO NORMAL E
INTRUSIVO DEL TRÁFICO EN UNA RED TCP/IP**

MARIO FERNANDO GARCÍA GUERRA

SANGOLQUÌ – ECUADOR

2011

CERTIFICACIÓN

Certificamos que el presente Proyecto de Grado Titulado ANÁLISIS DEL COMPORTAMIENTO NORMAL E INTRUSIVO DEL TRÁFICO EN UNA RED TCP/IP, ha sido desarrollado en su totalidad por el señor MARIO FERNANDO GARCÍA GUERRA con CC 0603405404, bajo nuestra dirección.

Atentamente

Ing. Carlos Romero

DIRECTOR

Ing. Marcelo Núñez

CODIRECTOR

RESUMEN

El presente proyecto determina el comportamiento normal del tráfico que circula por la red del Departamento de Eléctrica y Electrónica de la ESPE, mediante un modelo que basado en un análisis estadísticos pronostique la actividad de operación normal de la red, la finalidad principal es detectar nuevos y desconocidos ataques que intenten vulnerar las configuraciones de los protocolos TCP/IP. Para lo cual se ha estudiado la configuración y modo de operación de cada protocolo de la capa de Red (IP, ICMP) y de la capa de transporte (TCP, UDP) con el propósito de conocer la manera en que se manipulan a dichos protocolos, valiéndose de sus vulnerabilidades para lograr producir ataques causando así anomalías en la red.

La determinación del modelo de tráfico se la ha realizado mediante una captura de los paquetes que circularon durante una semana, con lo cual mediante software de análisis y monitoreo se ha logrado extraer los principales parámetros que rigen el comportamiento normal del tráfico TCP/IP. Además con el objetivo de conocer como se producen ataques hacia una red de estas características se generaron diferentes tipos de ataques, utilizando una red de pruebas, determinando así las principales características que estos presentan, facilitando al administrador reconocer de manera oportuna y tomar las medidas necesarias ante la presencia de uno de ellos.

DEDICATORIA

A la Virgen Dolorosa por darme la sabiduría y la fuerza para superar los momentos difíciles.

A mis padres por su amor incondicional, comprensión y ejemplo para nunca darme por vencido y alcanzar mis metas.

Mario Fernando García Guerra

AGRADECIMIENTO

A mi familia, por su valiosa ayuda, consejos y apoyo que me han brindado.

A la Escuela Politécnica del Ejército y a sus maestros por fomentar en mí una cultura de investigación y desarrollo en el campo de las telecomunicaciones.

Al Ing. Carlos Romero y al Ing. Marcelo Nuñez por su guía, tiempo y ayuda en el desarrollo de este proyecto.

A mis amigos y compañeros, por todo su apoyo, paciencia y amistad durante esta vida universitaria.

A todas las personas que me apoyaron de una u otra manera en mi superación personal y académica.

Mario Fernando García Guerra

PRÓLOGO

En el presente proyecto de tesis se estudió los problemas de seguridad en las redes TCP/IP, al determinar las vulnerabilidades en sus protocolos básicos, para posteriormente analizar el comportamiento normal e intrusivo del tráfico en una red TCP/IP logrando constituir así una de las armas fundamentales para vigilar los sucesos producidos en la red y detectar anomalías e intrusiones que pudieran estar afectando la misma.

Además se implementó una red de pruebas, para generar ataques, capturando y analizando los paquetes que circulan y determinar las características más comunes que se tiene ante una intrusión.

Se utilizó software y herramientas que permiten supervisar la red de estudio y generar los ataques más comunes. Se contó con el software FLUKE NETWORK PROTOCOL EXPERT, y el analizador de tramas WIRESHARK, con los cuales se pudo capturar cada paquete y determinar los parámetros normales y las anomalías de los principales encabezados de la familia de protocolos TCP/IP.

ÍNDICE GENERAL

CAPÍTULO 1

INTRODUCCIÓN

1. ANTECEDENTES	21
2. ALCANCE.....	22
3. OBJETIVOS	23
3.1 General	23
3.2 Específicos.....	23

CAPÍTULO 2

FUNDAMENTOS DE TCP/IP

1. TCP/IP	25
2. MODELO TCP/IP	26
2.1 Capa Acceso.....	27
2.2 Capa de Red.....	27
2.3 Capa de Transporte	27
2.4 Capa de Aplicación	28
3. PROTOCOLOS TCP/IP.....	28
4. PROTOCOLO TCP (TRANSMISSION CONTROL PROTOCOL)	30
4.1 Puertos y Sockets	30
4.1.1 Puertos Utilizados Normalmente.	30
4.2 Formato de la Cabecera TCP	31
4.2.1 Puerto de Origen.	32
4.2.2 Puerto de destino.	32
4.2.3 Numero de Secuencia.	32
4.2.4 Número de Confirmación.....	35

4.2.5	Comienzo de datos.....	35
4.2.6	Flags (URG, ACK, PSH, RST, SYN, FIN).	36
4.2.6.1	Flag URG (Urgent).....	36
4.2.6.2	Flag ACK (Acknowledge).....	36
4.2.6.3	Flag PSH (Push).....	36
4.2.6.4	Flag RST (Reset).....	37
4.2.6.5	Flag SYN (Synchronization).	37
4.2.6.6	Flag FIN (Finish).....	38
4.2.7	Ventana.	38
4.2.8	Suma de Comprobación.....	39
4.2.9	Puntero de Urgencia.....	40
4.2.10	Opciones.	40
4.3	Estados de Conexión TCP.....	41
4.3.1	Establecimiento de Conexión.	41
4.3.2	Cierre de Conexión.....	42
4.3.3	Lista de estados TCP.	43
4.3.3.1	LISTEN.....	43
4.3.3.2	SYN_SENT.....	43
4.3.3.3	SYN RECEIVED.....	44
4.3.3.4	ESTABLISHED.....	44
4.3.3.5	FIN WAIT_1.....	44
4.3.3.6	FIN WAIT_2.....	44
4.3.3.7	CLOSE WAIT.	44
4.3.3.8	CLOSING.	44
4.3.3.9	LAST ACK.	44
4.3.3.10	TIME_WAIT.....	45
5.	PROTOCOLO DE INTERNET (IP).....	45
5.1	Generalidades.....	45
5.2	Formato de la Cabecera IP.....	47
5.2.1	Versión: 4bits.....	47
5.2.2	Longitud de la cabecera IP (IHL): 4bits.	47
5.2.3	Tipo de Servicio (TOS): 8bits.	48
5.2.4	Longitud Total (Total Length): 16 bits.....	50

5.2.5	Identificación: 16 bits.....	50
5.2.6	Indicadores (flags): 3bits	50
5.2.7	Desplazamiento del fragmento (Fragment Offset): 13 bits.	51
5.2.8	Tiempo de Vida (TTL: Time to Live): 8bits.....	51
5.2.9	Protocolo: 8bits.....	51
5.2.10	Suma de Comprobación de la cabecera (Header Checksum): 16 bits.	52
5.2.11	Dirección IP de origen (Source Address): 32 bits.....	52
5.2.12	Dirección IP de destino (Destination Address): 32 bits.....	52
5.2.13	Opciones (Options).....	52
5.2.14	Padding.	54
5.3	Principio de Funcionamiento del Protocolo IP.....	55
5.3.1	Diferencia entre Paquete, Segmento y Datagrama.	55
5.4	Descripción de Funciones del Protocolo IP.....	59
5.4.1	Direccionamiento.....	60
5.4.2	Fragmentación.	61
5.4.2.1	Tráfico Fragmentado.	61
5.4.2.2	Teoría de la fragmentación.....	61
6.	PROCOLO ICMP.....	65
6.1	Solicitud y respuesta de eco	67
6.2	Mensajes ICMP de tiempo excedido.....	68
7.	SEGURIDAD TCP/IP.....	70
7.1	Vulnerabilidades de la capa de Acceso.	70
7.2	Vulnerabilidades de la capa de Red.	71
7.3	Vulnerabilidades de la capa de Transporte.....	71
7.4	Vulnerabilidades de la Capa de Aplicación.....	72
7.4.1	Servicio de nombres de dominio (DNS).	72
7.4.2	Telnet.	73
7.4.3	File Transfer Protocol (FTP).....	73
7.4.4	Hypertext Transfer Protocol (HTTP).....	74

CAPÍTULO 3

ANÁLISIS DEL TRÁFICO TCP/IP DE LA RED DE PRUEBAS DEL DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA DE LA ESPE (DEEE)

1. INTRODUCCIÓN.....	75
2. GENERALIDADES DE LA RED DE ESTUDIO	76
2.1 Arquitectura de la Red	76
2.1.1 Aspectos Estructurales.....	77
2.1.1.1 Conexión a Internet	77
2.1.1.2 Recursos disponibles	77
2.1.1.3 Modelo Genérico de la red	77
2.1.1.4 Elementos de monitorización	78
2.2 Configuración.....	78
2.2.1 Repetición de los patrones obtenidos.	78
2.2.2 El tamaño de los datos en disco.....	78
2.2.3 Seguridad.....	79
2.3 Herramientas de Monitoreo y Seguridad	79
2.3.1 Plataforma de funcionamiento.....	79
2.3.2 Licencia de uso	79
2.3.3 Continuidad del proyecto.....	80
2.3.4 Madurez del software	80
2.4 Herramientas Utilizadas.....	80
2.4.1 TCPDUMP.....	80
2.4.1.1 Utilización Básica de TCPDUMP.....	81
2.4.1.2 Utilización de TCPDUMP para almacenar datos de contenido completo.....	82
2.4.1.3 Utilización TCPDUMP para leer datos de contenido almacenados.	83
2.4.2 TCPREPLAY.....	84
2.4.2.1 Utilización TCPREPLAY para reproducir paquetes capturados con TCPDUMP.....	85
2.4.3 Fluke Network Protocol Expert.....	87
2.4.3.1 Características.....	87

2.4.3.1.1	Detección exhaustiva de dispositivos.....	88
3.	ANALISIS DE RESULTADOS	88
3.1	Análisis del primer día (miércoles 3 de octubre).	90
3.1.1	Interfaz eth0	90
3.1.1.1	Iniciar el programa Fluke Network Protocol Expert.	90
3.1.1.2	Inicio del proceso Monitoreo / Captura.....	91
3.1.1.3	Detener el proceso de captura.	95
3.1.1.4	Estimación de características y parámetros del tráfico capturado.	99
3.1.1.5	Distribución de Paquetes de acuerdo al tamaño.	101
3.1.1.6	Captura de Paquetes según la dirección MAC.....	103
3.1.1.7	Captura de Paquetes según la dirección IP.	106
3.1.1.8	Numero de paquetes que circulan de acuerdo a la dirección IP.	107
3.1.1.9	Numero de bytes que circulan de acuerdo a la dirección IP.....	108
3.1.1.10	Tiempo de respuesta.	108
3.1.1.11	Resumen de toda la actividad generada por la interfaz eth0 ...	110
3.1.2	Interfaz eth1	118
3.1.2.1	Cantidad de Bytes por Protocolo.....	118
3.1.2.2	Cantidad de Bytes	119
3.1.2.3	Distribución de Paquetes de acuerdo al tamaño	119
3.1.2.4	Captura de Paquetes según la dirección MAC.....	120
3.1.2.5	Captura de Paquetes según la Dirección IP	123
3.1.2.6	Numero de paquetes que circulan de acuerdo a la dirección IP .	123
3.1.2.7	Numero de bytes que circulan de acuerdo a la dirección IP.....	124
3.1.2.8	Tiempo de Respuesta	124
3.1.2.9	Resumen de toda la actividad generara por la interfaz eth1	125
3.1.3	Interfaz eth2	129
3.1.3.1	Cantidad de Bytes por Protocolo.....	129
3.1.3.2	Cantidad de Bytes	130
3.1.3.3	Distribución de Paquetes de acuerdo al tamaño	131
3.1.3.4	Captura de Paquetes según la dirección MAC.....	132
3.1.3.5	Captura de Paquetes según la Dirección IP	133
3.1.3.6	Numero de paquetes que circulan de acuerdo a la dirección IP .	134
3.1.3.7	Numero de bytes que circulan de acuerdo a la dirección IP.....	134

3.1.3.8	Tiempo de Respuesta	135
3.1.3.9	Resumen de toda la actividad generada por la interfaz eth2.....	135

CAPÍTULO 4

ANÁLISIS DEL TRÁFICO DE RED TCP/IP SOBRE UNA RED DE PRUEBAS

1.	IMPLEMENTACIÓN DE LA RED	141
1.1	Herramientas utilizadas.....	142
1.1.1	NMAP (Network Mapper).	142
2.	ACTIVIDADES Y MÉTODOS MÁS COMUNES DE ATAQUE.....	143
2.1	Actividades previas a la realización de un ataque	143
2.1.1	Utilización de herramientas de administración.	143
2.1.1.1	La herramienta Ping	143
2.1.1.2	Traceroute.	144
2.1.1.3	Netstat.	144
2.1.1.4	Whois	146
2.1.1.5	Nslookup	146
2.1.2	Exploración de puertos.....	148
2.1.2.1	Exploración de puertos TCP.....	148
2.1.2.2	Exploración de puertos UDP.	149
2.1.2.3	Herramientas para realizar la exploración de puertos.	150
2.1.2.4	Análisis de resultados.....	150
2.1.3	Escuchas de red (Sniffers).	154
3.	TIPOS DE ATAQUES TCP/IP	154
3.1	Denegación de Servicio.	154
3.1.1	Ataques de denegación de servicio distribuidos.....	155
3.1.1.1	TRINOO.	156
3.1.1.2	TFN (Tribe Flood Network).....	158
3.1.1.3	TFN2K.....	159
3.1.1.4	Shaft.....	160
3.2	Tipos de ataque DOS (Denegación de Servicio)	161
3.2.1	Ataque por fragmentación.	161

3.2.2	IP Flooding.....	163
4.	GENERACIÓN DE DIFERENTES TIPOS DE ATAQUES.....	164
4.1	Spoofing.....	164
4.2	TCP/SYN <i>Flooding</i>	165
4.3	Análisis del Tráfico Generado.....	166
4.4	Smurf.....	170
4.5	Análisis del Tráfico Generado.....	172
4.6	Teardrop.....	175
4.7	Ping of death.....	175
4.8	Análisis del Tráfico Generado.....	176

CAPÍTULO 5

ANÁLISIS ESTADÍSTICO DE LOS DATOS DEL TRÁFICO TCP/IP DE LA RED DEL DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA DE LA ESPE (DEEE)

1.	INTRODUCCIÓN.....	179
2.	CONCEPTOS ESTADÍSTICOS.....	180
2.1	Selección de la muestra.....	180
2.1.1	Población.....	180
2.1.2	Muestreo.....	181
2.2	Técnicas del muestreo.....	182
2.2.1	Muestreo probabilístico.....	182
2.2.2	Muestreo no probabilístico.....	182
2.2.3	Tamaño de la muestra.....	182
3.	ESTADÍSTICA DESCRIPTIVA.....	183
3.1	Utilización de herramientas de office 2007.....	183
3.1.1	Herramientas / Análisis de datos.....	184
3.2	Estadígrafos de posición.....	187
3.2.1	Medidas tendencia central.....	187
3.2.1.1	Media.....	187
3.2.1.2	Mediana.....	189
3.2.1.3	Moda.....	189

3.2.1.4	Resumen.....	190
3.2.2	Medidas de Dispersión.....	190
3.2.2.1	Varianza	190
3.2.2.2	Desviación estándar o Típica	192
3.2.3	Medidas de Distribución.....	192
3.2.3.1	Asimetría.....	192
3.2.3.2	Curtosis.....	194
3.3	Correlación y Regresión Lineal.....	196
3.3.1	Correlación Lineal.....	196
3.3.2	Definición y características del concepto de Regresión Lineal....	197
3.3.2.1	Definición del Coeficiente de Determinación.....	197
4.	ANÁLISIS ESTADÍSTICOS DE LOS DATOS DEL TRÁFICO TCP/ IP DE LA RED DEL DEEE.....	198
4.1	Análisis del Comportamiento diario.....	198
4.1.1	Análisis de los Datos del Tráfico de la Red del DEEE.....	199
4.1.2	Cálculo del Tamaño de la Muestra.....	200
4.1.3	Aplicación de Muestreo (Herramientas / Análisis de Datos).....	201
4.1.4	BUSCARV (Valor que se desea buscar en la matriz; Matriz de datos donde buscar datos; Columna que se desea obtener dato; Ordenado). ..	202
4.1.5	Análisis: Miércoles mañana primera semana (eth0).....	203
4.1.5.1	Cantidad de Bytes por segundo.....	203
4.1.5.2	Tiempo entre arribos de tramas.....	205
4.1.5.3	Desarrollo del Modelo de Tráfico.....	210
4.1.5.4	Extracción de la Serie.....	210
4.1.6	Identificación del Modelo.....	212
4.1.6.1	Definición y características del concepto de Regresión Lineal....	212
4.1.6.2	Interpretación de los coeficientes estimados.....	214
4.1.6.3	Inferencia en el modelo de regresión.....	215
4.1.6.4	Análisis de regresión.....	217
4.1.7	Análisis: Miércoles tarde, primera semana (eth0).....	218
4.1.7.1	Comportamiento de la variable y.....	218
4.1.7.2	Comportamiento de ambas variables.....	218
4.1.7.3	Análisis estadígrafos.....	219
4.1.7.4	Análisis de regresión.....	219

4.1.8	Análisis: Miércoles mañana segunda semana (eth 0)	221
4.1.8.1	Comportamiento de la variable y	221
4.1.8.2	Comportamiento ambas variables	222
4.1.8.3	Análisis estadígrafos	222
4.1.8.4	Análisis de regresión	223
4.1.9	Análisis: Miércoles tarde, segunda semana (eth 0)	224
4.1.9.1	Comportamiento de la variable y	224
4.1.9.2	Comportamiento de ambas variables	224
4.1.9.3	Análisis estadígrafos	225
4.1.9.4	Análisis de regresión	226
4.2	Análisis comportamiento todo el período	227
4.2.1	Cantidad de bits	227

CONCLUSIONES Y RECOMENDACIONES.....	233
REFERENCIAS BIBLIOGRÁFICAS.....	2389
GLOSARIO.....	242
ANEXOS.....	244

ÍNDICE DE FIGURAS

Figura. 2.1. Modelo TCP/IP	26
Figura. 2.2. Protocolos y capas TCP/IP	29
Figura. 2.3. Número de Secuencia.....	33
Figura. 2.4. Número de Confirmación	35
Figura. 2.5. Tamaño de Ventana.....	38
Figura. 2.6. Establecimiento de Conexión	42
Figura. 2.7. Cierre de Conexión	43
Figura. 2.8. Formato de la cabecera IP	47
Figura 2.9 Estructura del Campo TOS.....	49
Figura. 2.10. Modelo de Operación	55
Figura. 2.11. Modelo de Operación para transmitir un Datagrama de una Aplicación a otra.....	56
Figura. 2.12. Datagrama Formado	57
Figura. 2.13. Paquete Enviado	57
Figura. 2.14. Datagrama como lo Genera la Capa IP	58
Figura. 2.15. Datagrama para trasmitirlo	58
Figura. 2.16. Modelo de Operación para transmitir un Datagrama a la red local B	59
Figura. 2.17. Modelo de Operación para transmitir un Datagrama a una capa de Aplicación.....	59
Figura. 2.18. Datagrama Ethernet (MTU=1500).....	63
Figura. 2.19. Fragmento de 4028 bytes dividido en dos fragmentos de 1500 bytes y uno de 1068 bytes.	63
Figura. 2.20. Fragmento inicial	64
Figura. 2.21. Segundo Fragmento.....	64
Figura. 2.22. Último Fragmento.....	65
Figura. 2.23. Orden PING.....	67
Figura. 3.1. Red del departamento de Eléctrica y Electrónica de la ESPE	76
Figura. 3.2. Inicio del Programa Fluke Network Protocol Expert	91
Figura. 3.3. Inicio del Proceso Monitoreo/Captura	92
Figura. 3.4. Tramas capturadas y recibidas de Broadcast y Multicast	93

Figura. 3.5. Dirección MAC y Modelo de la NIC	93
Figura. 3.6. Ventana Detail View	94
Figura. 3.7. Ventana de Captura	95
Figura. 3.8. Valores HEX del paquete	96
Figura. 3.9. Código de colores para la ubicación	97
Figura. 3.10. EtherType (0x0800) Paquete IP	97
Figura. 3.11. Direcciones IP y datos almacenados en la vista HEX	98
Figura. 3.12. Información del User Datagram Protocol	98
Figura. 3.13. Interfaz eth0 mañana	99
Figura. 3.14. Cantidad de bytes por Protocolo	100
Figura. 3.15. Distribución de paquetes de acuerdo al tamaño.....	103
Figura. 3.16. Captura de paquetes según la dirección MAC	105
Figura. 3.17. Número de paquetes que circulan de acuerdo a la dirección IP ...	107
Figura. 3.18. Número de bytes que circulan de acuerdo a la dirección IP	108
Figura. 3.19. Cantidad de bytes por protocolo.....	119
Figura. 3.20. Distribución de paquetes de acuerdo al tamaño.....	120
Figura. 3.21. Captura de paquetes según la Dirección MAC	122
Figura. 3.22. Número de paquetes que circulan de acuerdo a la Dirección IP...	123
Figura. 3.23. Número de bytes que circulan de acuerdo a la Dirección IP	124
Figura. 3.24. Cantidad de bytes por protocolo.....	130
Figura 3.25. Distribución de paquetes de acuerdo al tamaño	131
Figura 3.26. Captura de paquetes según la dirección MAC	132
Figura. 3.27. Número de paquetes que circulan de acuerdo a la Dirección IP...	134
Figura. 4.1. Red de Pruebas Implementada.....	141
Figura. 4.2. Ataque ejecutado mediante <i>TRINOO</i>	157
Figura. 4.3. Esquema de una conexión Telnet.....	158
Figura. 4.4. Esquema de comunicaciones de Shaft	161
Figura. 4.5. Inundación de paquetes SYN de TCP.....	167
Figura. 4.6. Ataque <i>Smurf</i>	171
Figura. 4.7. Generación de Ataque <i>Smurf</i>	171
Figura. 5.1. Estados de Asimetría	193
Figura. 5.2. Concentración de Valores	194
Figura. 5.3. Distancia de dos desviaciones estándar de la media aritmética ...	195

Figura. 5.4. Distribución de las cantidades de bytes arribadas por segundo .	204
Figura 5.5. Variación en función de la cantidad de bites	205
Figura. 5.6. Gráfico de los tiempos entre arribo de las 381 tramas observadas.	206
Figura. 5.7. Porcentaje de Tráfico de cada Protocolo.....	208
Figura. 5.8. Tráfico TCP, ICMP, UDP.....	209
Figura. 5.9. Serie de Tiempo del Tráfico Capturado.....	211
Figura. 5.10. Tráfico Real.....	213
Figura. 5.11. Gráfico de los Residuales	217
Figura. 5.12. Comportamiento de la Variable “y”.....	218
Figura. 5.13. Comportamiento de Ambas Variables.....	218
Figura. 5.14. Variable X1 Gráfico de los residuales	220
Figura. 5.15. Comportamiento de la Variable y	221
Figura. 5.16. Comportamiento de Ambas Variables.....	222
Figura. 5.17 Variable X1 Gráfico de los Residuales	223
Figura. 5.18. Comportamiento de la Variable y	224
Figura. 5.19. Comportamiento de Ambas Variables.....	224
Figura. 5.20. Variable X1 Gráfico de los residuales	226
Figura. 5.21. Comportamiento de la cantidad de paquetes que circularon en la red por días, por cada estación.	229
Figura. 5.22. Comportamiento de la cantidad de paquetes que circularon en la red por la mañana y la tarde.....	231
Figura. 5.23. Comportamiento de la Circulación en la red	232

ÍNDICE DE TABLAS

Tabla. 2.1. Puertos utilizados Normalmente.....	31
Tabla. 2.2. Formato de cabecera TCP	32
Tabla.2.3. El Campo TOS	48
Tabla. 2.4. Segunda Parte del TOS	49
Tabla. 2.5. Indicadores (flags): 3 bits	50
Tabla. 2.6. Estructura del Option-type.....	53
Tabla. 2.7. Option class.....	53
Tabla. 2.8. Option Number	54
Tabla. 2.9. Protocolo ICMP	66
Tabla. 3.1. Resumen de la cantidad de bytes por protocolo	101
Tabla.3.2 Distribución de paquetes de acuerdo al tamaño.....	102
Tabla. 3.3. Captura de paquetes según la dirección MAC	104
Tabla. 3.4. Captura de paquetes según la dirección IP.....	106
Tabla. 3.5. Tiempo promedio utilizado para cada conexión	109
Tabla. 3.6. Resumen de toda la actividad generada por la interfaz eth0.....	110
Tabla. 3.7. Resumen de toda la actividad generada por la interfaz eth0.....	114
Tabla. 3.8. Resumen de toda la actividad generada por la interfaz eth0.....	115
Tabla. 3.9. Resumen de toda la actividad generada por la interfaz eth0.....	117
Tabla. 3.10. Cantidad de bytes por protocolo.....	118
Tabla. 3.11. Distribución de paquetes de acuerdo al tamaño.....	119
Tabla. 3.12. Captura de paquetes según la dirección MAC	121
Tabla. 3.13. Captura de paquetes según la Dirección IP	123
Tabla. 3.14. Tiempo de respuesta.....	124
Tabla. 3.15. Resumen de toda la actividad generada por la interfaz eth1.....	125
Tabla. 3.16. Resumen de toda la actividad generada por la interfaz eth1.....	126
Tabla. 3.17. Resumen de toda la actividad generada por la interfaz eth1.....	127
Tabla. 3.18. Resumen de toda la actividad generada por la interfaz eth1.....	128
Tabla. 3.19. Cantidad de bytes por protocolo.....	129
Tabla. 3.20. Captura de paquetes según la dirección IP.....	133
Tabla. 3.21. Tiempo de respuesta.....	135

Tabla. 3.22. Resumen de toda la actividad generada por la interfaz eth2.....	136
Tabla 3.23. Resumen de toda la actividad generada por la interfaz eth2.....	137
Tabla 3.24. Resumen de toda la actividad generada por la interfaz eth2.....	138
Tabla 3.25. Resumen de toda la actividad generada por la interfaz eth2.....	139
Tabla 3.26. Resumen de toda la actividad generada por la interfaz eth2.....	140
Tabla. 4.1. Tráfico que circula por la interfaz del host 10.1.1.2	152
Tabla. 4.2. Hosts activos en la subred	152
Tabla. 4.3. Verificación de Nmap	153
Tabla. 4.4. Tráfico capturado por wireshark en la interfaz del host 10.1.1.2	153
Tabla. 4.5. Tráfico que entra por la interfaz del servidor	169
Tabla. 4.6. Tráfico que circula por la interfaz.....	173
Tabla 4.7: Host 10.1.1.5	174
Tabla. 5.1. Ejemplo Datos Capturados.....	200
Tabla. 5.2. Ejemplo de Muestras Obtenidas	203
Tabla. 5.3. Análisis de los Diferentes Estadígrafos	216
Tabla. 5.4. Estadísticas de la Regresión	217
Tabla. 5.5. Análisis de los Diferentes Estadígrafos	219
Tabla. 5.6. Estadística de la Regresión.....	220
Tabla. 5.7. Análisis Estadígrafos	223
Tabla. 5.8. Análisis de Regresión.....	223
Tabla. 5.9. Análisis Estadígrafos	225
Tabla. 5.10. Análisis de regresión	226
Tabla. 5.11. Cantidad de bits que circularon en la red en los 8 días	227

CAPÍTULO 1

INTRODUCCIÓN

1. ANTECEDENTES

Las redes de comunicación fueron creadas con el fin de compartir información y recursos, por ende las organizaciones son cada vez más dependientes de sus redes de datos siendo la información el activo más importante que una empresa puede tener, por tal motivo cualquier problema que las afecte, por mínimo que sea puede comprometer la continuidad de sus operaciones.

La seguridad de la red es a veces, menospreciada por sus administradores, poniendo en peligro la integridad y confidencialidad de una red de datos, pues un usuario no autorizado puede acceder fácilmente a servidores e información convirtiéndose en una amenaza.

Al existir ataques generados desde la misma red interna o externos desde cualquier conexión a internet, y que pretenden acceder ilegítimamente a sistemas informáticos ya sean para sustraer, modificar o eliminar información y afectar el funcionamiento de uno o varios servicios, es de gran importancia para el administrador conocer el comportamiento normal del tráfico de la red operativa con el fin de detectar posibles intrusiones y tomar las medidas pertinentes.

Conociendo el riesgo de vulnerabilidad que puede existir en una red TCP/IP, las capturas de tráfico para su posterior análisis han permitido, determinar los parámetros normales y anormales de los principales encabezados de la familia de protocolos TCP/IP, y determinar los riesgos de seguridad a los que eventualmente está sometida la red

2. ALCANCE

El proyecto de investigación comprenderá el análisis del comportamiento normal e intrusivo del tráfico de la red del Departamento de Eléctrica y Electrónica de la ESPE permitiendo determinar las características y patrones de la red operativa, para esto se estudiará los conceptos básicos de TCP/IP, centrándose en detalle en los protocolos que conforman dicha arquitectura, analizando los diferentes tipos de ataque y las vulnerabilidades que presentan tanto la capa de red como de transporte.

Además se implementará una red de pruebas, para generar ataques, capturando y analizando los paquetes que circulan y determinar las características más comunes que se tiene ante una intrusión.

Se realizará una comparación entre las características que se presentan cuando se produce un ataque y las del comportamiento normal, con la finalidad de poder establecer los principales parámetros y diferencias que ocurren en ambos eventos.

Dado que al producirse algún tipo de ataque, estos presentan características que los permiten identificar en base al conocimiento del comportamiento normal de la red operativa, el estudio y análisis de estos parámetros se analizarán a profundidad en el desarrollo del proyecto.

Se utilizará software y herramientas que permitan supervisar la red de estudio y generar los ataques más comunes. Se contará con el software FLUKE

NETWORK INSPECTOR (el cual posee licencia la ESPE), y el analizador de tramas WIRESHARK, con los cuales se podrá capturar cada paquete y determinar los parámetros normales y las anomalías de los principales encabezados de la familia de protocolos TCP/IP

3. OBJETIVOS

3.1 General

Analizar el comportamiento normal e intrusivo del tráfico de una red TCP/IP, a partir de la captura y características de los protocolos que conforman la arquitectura TCP/IP.

3.2 Específicos

- Analizar los protocolos que conforman la arquitectura TCP/IP.
- Determinar los problemas de seguridad en las redes TCP/IP al analizar las vulnerabilidades en sus protocolos básicos
- Analizar los tipos de ataques, métodos y herramientas utilizados para generarlos
- Determinar las características y el comportamiento que presenta el tráfico de la red operativa de los laboratorios del Departamento de Eléctrica y Electrónica de la ESPE
- Implementar una red de pruebas, generar diferentes tipos de ataques y determinar el comportamiento que presentan los ataques más comunes producidos a nivel de red y de transporte

- Comparar los análisis obtenidos en los distintos tipos de comportamiento y establecer diferencias y patrones que permitan detectar posibles intrusiones

CAPÍTULO 2

FUNDAMENTOS DE TCP/IP

1. TCP/IP

La familia de protocolos TCP/IP (*Transport Control Protocol / Internet Protocol*) se originó alrededor de 1960 como base de un sistema de comunicación basado en redes de conmutación de paquetes desarrollado por la Agencia de Proyectos de Investigación Avanzada del Departamento de Defensa de los Estados Unidos ARPA. Actualmente constituye la infraestructura tecnológica más extendida y desarrollada sobre la que circulan las comunicaciones electrónicas (datos, voz, multimedia...). Su expansión se ha debido principalmente al desarrollo del Internet¹

Los protocolos TCP/IP son implementados en todos los sistemas y dispositivos de red y permiten su interconexión independientemente su función, plataforma y sistema operativo que estos posean, siendo todos ellos objetivo de potenciales ataques.

Los sistemas son los equipos que engloban tanto a los clientes de un servicio o comunicación, ya sean computadoras de escritorio o estaciones de trabajo, así como dispositivos móviles (PDAs, teléfonos móviles...), como a los

¹ Raúl Siles Peláez (Análisis de Seguridad de la Familia Protocolos TCP/IP)

servidores que proporcionan el servicio siendo estos últimos objetivo de los *hackers*, al contener información relevante.

Los dispositivos de red son los encargados de conectar dos o más redes y direccionar el tráfico entre las mismas. Por tanto engloban a las repetidoras, *bridges*, *hubs*, *switches*, *routers*, *firewalls*, servidores de terminales y acceso (RAS) etc.

2. MODELO TCP/IP

TCP/IP está diseñado en una estructura de capas, fundamentada en el estándar de los protocolos de comunicaciones que diseñó la organización ISO, denominado OSI (*Open Systems Interconnection*). Cada una de las capas es responsable de llevar a cabo una tarea específica para la comunicación.

TCP/IP se diferencia del modelo OSI en que solo tiene 4 capas: Capa Física, Capa de Red, Capa de Transporte y Capa de Aplicación²

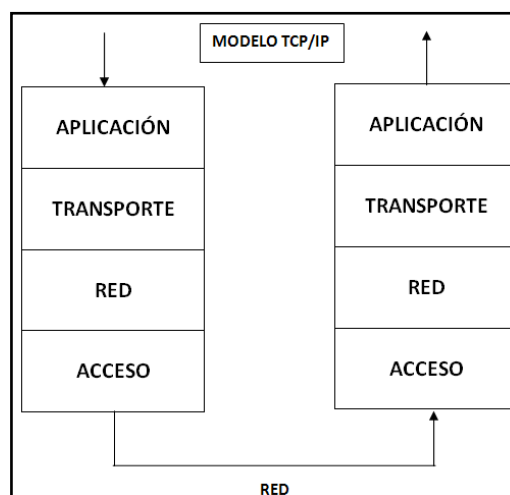


Figura. 2.1. Modelo TCP/IP

² Joaquín García Alfaro/ Xavier Perramón Tornil (Aspectos avanzados de Seguridad en Redes)

2.1 Capa Acceso

Asociada al medio de comunicación físico y de enlace, capas 1 y 2 de OSI, incluye los *drivers* que controlan las tarjetas de red y toda la gestión de la conexión entre el hardware y los cables y dispositivos de red. Todos los equipos conectados a internet implementan esta capa.

Esta capa se encarga de controlar el acceso al nivel físico utilizado y enviar la información por el mismo. Transforma a información básica (bits) toda la información que le llega de las capas superiores y la prepara para que se pueda enviar por el medio. El direccionamiento físico de la red lo hace mediante direcciones MAC.

Los protocolos asociados a este nivel no pertenecen propiamente a TCP/IP pero son la base sobre el que éste se desarrolla.

2.2 Capa de Red

Es la capa responsable de proporcionar conectividad entre usuarios, se encarga del envío y recepción de los paquetes a través de la red, así como de direccionarlos y encaminarlos hacia su destino, no proporciona fiabilidad en las conexiones, de esto ya se ocupa la capa de transporte. Realiza un direccionamiento lógico de red mediante las direcciones IP.

Principalmente el protocolo IP, se encarga de estas tareas.

2.3 Capa de Transporte

Se encarga del transporte de datos, el control de flujo y proporciona mecanismos de control de flujo y proporciona mecanismos de detección y

corrección de errores, sólo es implementada por equipos usuarios de internet o por terminales de internet. Los dispositivos de encaminamiento (routers) no la necesitan.

La información que le llega de la capa de aplicación la divide formando diferentes segmentos. El direccionamiento se realiza a través de puertos. Sus funcionalidades básicas son:

- Fiabilidad
- Control de flujo
- Corrección de errores
- Retransmisión

Existen dos protocolos a este nivel: TCP, un protocolo fiable y orientado a conexión, y UDP, un protocolo más simple pero que no garantiza la recepción de los datos, además, no orientado a conexión.

2.4 Capa de Aplicación

Es la capa más cercana al usuario final, proporciona servicios de red, es la responsable de traducir los datos de la aplicación, programa, para que puedan ser enviados por la red. Sus funciones se resumen en:

- Representación
- Codificación
- Control de diálogo

3. PROTOCOLOS TCP/IP

Para poder enviar información entre dos máquinas, es necesario que ambas estaciones hablen el mismo lenguaje para que se entiendan entre

ellas. A este lenguaje se le llamará *protocolo*. En cada una de las capas expuestas encontramos protocolos distintos

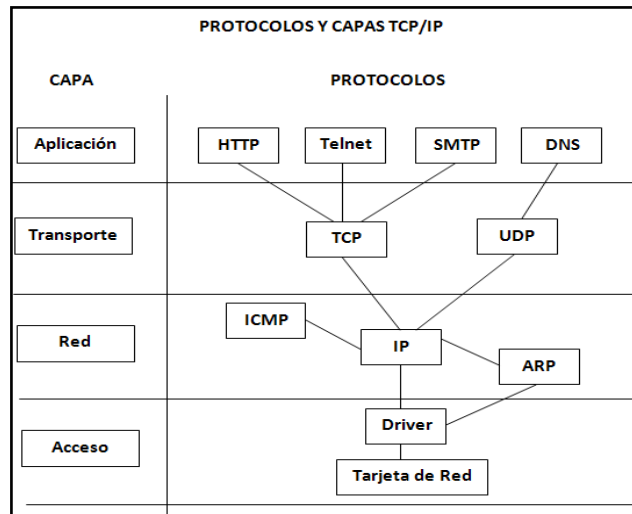


Figura. 2.2. Protocolos y capas TCP/IP

Los protocolos más representativos que figuran en la capa de aplicación son:

- File Transfer Protocol (FTP)
- Hypertext Transfer Protocol (HTTP)
- Simple Mail Transfer Protocol (SMTP)
- Domain Name System (DNS)
- Trivial File Transfer Protocol (TFTP)
-

Los protocolos de la capa de Transporte son:

- Transport Control Protocol (TCP)
- User Datagram Protocol (UDP)

El protocolo más conocido de la capa de Internet es:

- Internet Protocol (IP)

El protocolo utilizado en la mayoría de redes locales en la capa de acceso es:

- Ethernet

4. PROTOCOLO TCP (TRANSMISSION CONTROL PROTOCOL)

El protocolo TCP, es el empleado en la mayoría de los servicios que componen Internet actualmente. Es un protocolo fiable, es decir, se asegura de que los paquetes de datos llegan al otro extremo mediante el uso de números de secuencia y de confirmaciones de recepción (ACKs), y es orientado a conexión, es preciso que las dos partes que van a comunicarse conozcan a la otra y establezcan una conexión formal. Asimismo, esta debería terminarse de forma adecuada.

4.1 Puertos y Sockets

En la capa de Transporte se hace referencia a los protocolos de la capa superior mediante sus números de puerto, los cuales son números de 2 bytes que relacionan a una aplicación o proceso en particular. Los puertos reservados son aquellos que tienen un valor del 1 al 1023, estos puertos se asignan a las aplicaciones a través de la IANA (Internet Assigned Numbers Authority). Como la utilización de estos puertos está controlada por un núcleo de estándares, a veces se les llama "puertos bien conocidos".

Los puertos del intervalo de 1024 a 65535 se llaman puertos registrados, aunque están enumerados por la IANA, no están estandarizados.

4.1.1 Puertos Utilizados Normalmente.

Se los presenta en la tabla siguiente:

Puerto/Protocolo	Nombre de servicio
7	Echo
9	Discard
15	Netstat
20	ftp-data
21	ftp
22	Ssh
23	telnet
25	Sntp
53	Domain
69	Tftp
79	Finger
80	www
110	pop3
113	Auth
137	netbios-ns
161	Snmp
220	imap3
443	https
512	Exec
513	Login
520	Route

Tabla. 2.1. Puertos utilizados Normalmente

Un número de puerto asociado a una dirección IP se le denomina socket

4.2 Formato de la Cabecera TCP

A continuación en la tabla se describe un formato de este tipo:

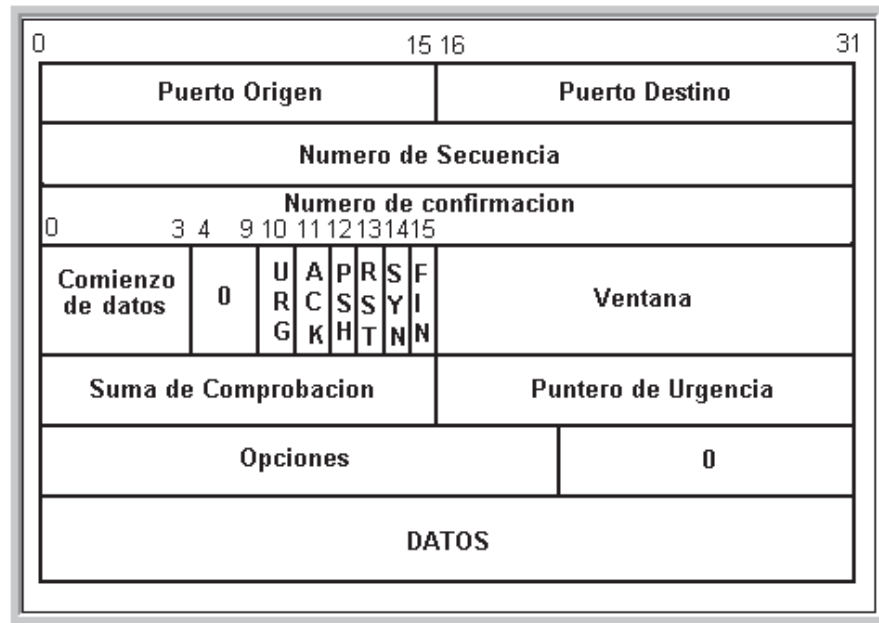


Tabla. 2.2. Formato de cabecera TCP

4.2.1 Puerto de Origen.

Este campo tiene el mismo significado que en el protocolo UDP, es decir es de condición necesaria para identificar una conexión virtual.

4.2.2 Puerto de destino.

Al igual que en el puerto de origen sirve para identificar una conexión virtual.

4.2.3 Número de Secuencia.

Por una parte 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. Durante un margen de tiempo (que depende del flujo de datos y, por lo tanto del ancho de banda) se tiene la garantía de que

en una determinada conexión virtual no existirá dos paquetes diferentes con el mismo número de secuencia.

El valor del número de secuencia 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.

Cuanto se tarda en dar la vuelta al marcador depende directamente de cuantos datos se envíen, y en cuanto tiempo³

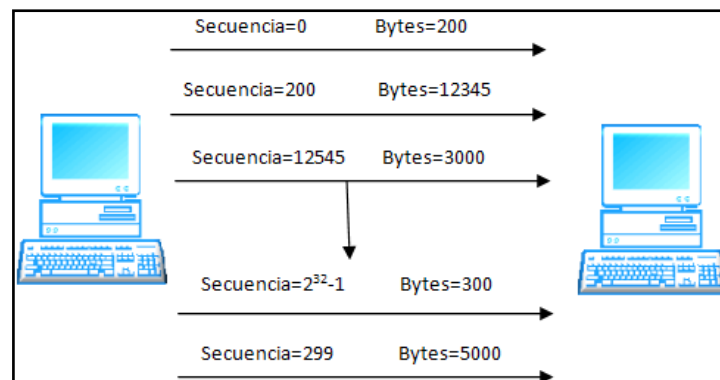


Figura. 2.3. Número de Secuencia

Como vemos, 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.

³ HACK X CRACK No 20 (TCP- La esencia de las comunicaciones por Red)

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 contamos 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 quedaría: 299 que sería el número de secuencia del próximo paquete.

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

De manera que una conexión identifica 5 parámetros: ip de origen, ip de destino, puerto de origen, puerto de destino, y número de secuencia.

Por tanto, el único parámetro con el que se puede contar para poder establecer nuevas conexiones similares sin que se confunda la nueva con las anteriores que ya se cerraron es el número de secuencia.

Si cada vez que se crea una conexión de un host a otro en la cual se utiliza el mismo número de secuencia para comenzar la sesión (por ejemplo el 0), si llegase un paquete retrasado de una conexión anterior, no habría forma de saber si este paquete pertenecía a la sesión anterior o a la nueva sesión. Así entonces una forma sencilla de evitar esto es no utilizar siempre el mismo número de secuencia para comenzar una sesión, si no ir utilizando números de secuencia cada vez mayores y así detectar paquetes duplicados, diferenciar unas conexiones de otras y ordenar los paquetes.

4.2.4 Número de Confirmación.

Este campo es utilizado para hacer confirmaciones de recepción, su funcionamiento es muy similar al del número de secuencia, porque también representa un orden en bytes de los datos.

El número de confirmación es otro número de 32 bytes que indica cual es el próximo byte que se espera recibir.

Si la otra parte en una conexión envía un paquete con número de secuencia 200. Y el paquete contiene 12345 bytes de datos, entonces el próximo número de secuencia sería el 12545. Para confirmar que se ha recibido el paquete con número de secuencia 200 se debe enviar un paquete de respuesta con el número de confirmación 12545, confirmando la llegada del paquete anterior y esperando el nuevo paquete que deberá tener numero de secuencia 12545.

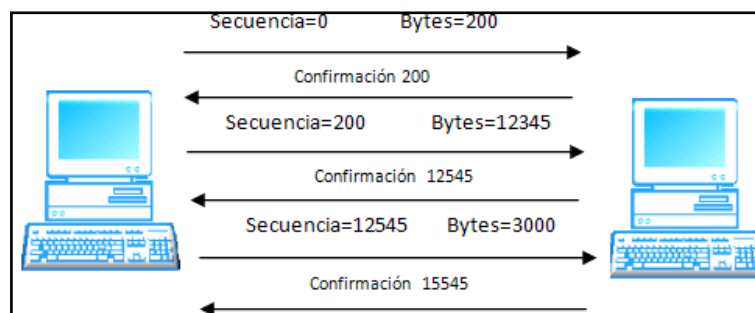


Figura. 2.4. Número de Confirmación

4.2.5 Comienzo de datos.

Este campo no es necesario en el caso de UDP ya que la cabecera UDP tiene un tamaño fijo de 8 bytes, en cambio la cabecera TCP puede tener un tamaño variable, por lo que es necesario indicar en 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, lo mínimo que puede ocupar la cabecera TCP.

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

Los flags son una serie de indicadores de control, de un único bit cada uno, con diferentes funciones

4.2.6.1 Flag URG (Urgent).

Cuando este flag está activo “1”, se indica que el paquete contiene datos urgentes y así estos datos serán procesados con mayor prioridad.

4.2.6.2 Flag ACK (Acknowledge).

Cuando el flag esta activo “1” significa que el paquete aparte de los datos propios, contiene una confirmación de respuesta a los paquetes que está enviando al otro extremo de la conexión.

Por tanto, el campo **numero de confirmación** no tendría significado al no estar activo este flag.

4.2.6.3 Flag PSH (Push).

Cuando este flag esta activo “1” indica al sistema tanto local como remoto que deben vaciar sus buffers de transmisión y recepción.

En todo sistema TCP tiene que haber dos pequeñas memorias o buffers: una para transmisión, y otra para recepción. Estos buffers son unas colas en las que se van almacenando los paquetes que hay que procesar.

4.2.6.4 Flag RST (Reset).

Cuando este flag esta activo "1", indica al otro extremo de la conexión que los datos que han llegado no coinciden con la conexión, por lo que se ha perdido la sincronización entre ambas partes.

Ante cualquier campo incorrecto que se reciba (números de secuencia inválidos, o flags no esperados) se tiene que responder con un paquete con este flag activo, para que el otro extremo se entere del problema, y se cierre la conexión con el fin de re-sincronizar ambas partes.

Si por ejemplo se envía varios paquetes para establecer una conexión, y al final uno de ellos tiene éxito y después de ese paquete de conexión llegan el resto de intentos de, se obtendrá como respuesta un RST a cada nuevo intento de conexión, con el fin de mantener la conexión ya establecida.

4.2.6.5 Flag SYN (Synchronization).

Cuando esta activo este flag "1", indica al otro extremo que se desea establecer una nueva conexión. Este flag se utiliza únicamente al abrir una nueva conexión.

4.2.6.6 Flag FIN (Finish).

Cuando este flag esta activo "1", indica al otro extremo de la conexión que la conexión ya se puede cerrar.

Tanto RST como FIN se utilizan para finalizar conexiones, pero la diferencia es que RST avisa de una situación de error, mientras que FIN avisa de una terminación sin problemas.

4.2.7 Ventana.

Este campo es el que se utiliza para llevar a cabo el control de flujo implementado en TCP.

El control de flujo permite evitar la congestión debida a la diferencia de velocidad (por capacidad de procesamiento, o por ancho de banda) entre ambas partes de una conexión.

Normalmente, el tamaño de la ventana está relacionado con la cantidad de espacio libre del buffer de recepción.

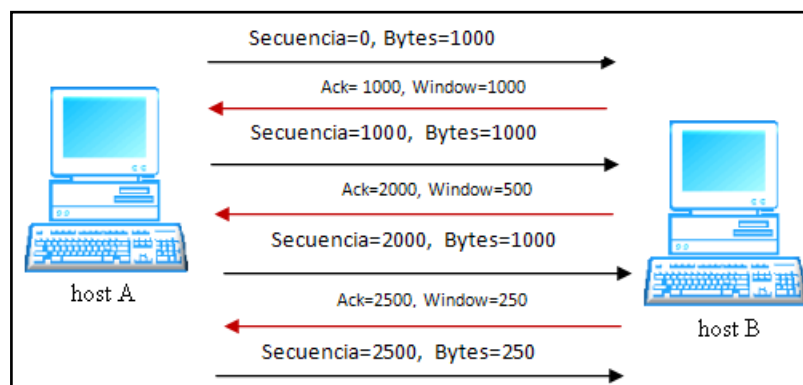


Figura. 2.5. Tamaño de Ventana

El host A envía un paquete de 1000 bytes, con un número de secuencia 0. El host B lo recibe y procesa, y envía su confirmación de recepción el

número de 1000 y además indica que su buffer puede recibir otros 1000 bytes.

El host A transmite 1000 bytes más de datos empezando por el número de secuencia de 1000. El host B recibe los datos, sin embargo se reduce su capacidad de procesamiento y negocia una longitud máxima de datos de la próxima transferencia de 500 bytes.

El host A, por algún motivo no recibe a tiempo la nueva ventana y seguirá enviando los paquetes al ritmo que llevaba anteriormente. Por lo tanto, el próximo paquete (secuencia= 2000, y longitud de datos =1000 bytes), a pesar de que el buffer de B solo admite 500 bytes.

Después de el envió, A recibe el aviso de B de congestión, por lo que decide esperar, cuando B procesa algunos datos, manda un paquete de confirmación Ack=2500, lo que indica que solo se proceso 500 bytes de los 1000 que se le envió y, además, el próximo paquete tendrá que ser como máximo de 250 bytes.

En respuesta A envía un nuevo paquete, que contenga 250 bytes.

4.2.8 Suma de Comprobación.

La suma de comprobación se calcula mediante una operación aritmética binaria que se realiza sobre una cabecera especial que contiene los datos más relevantes.

Cada vez que se recibe un paquete TCP, se realiza esta operación con los datos recibidos y se compara el número obtenido con el campo suma de comprobación del paquete, si ambos números no son iguales, los datos recibidos son incorrectos y será necesaria una retransmisión de los datos. En

este caso no es enviado ningún paquete de confirmación, por lo que el remitente deberá transmitir nuevamente el paquete.

4.2.9 Puntero de Urgencia.

TCP permite combinar en un mismo paquete datos urgentes con datos no urgentes, esto solo será permitido si el flag URG está activo, si no es así simplemente el campo puntero de urgencia será ignorado.

Este puntero es de importancia cuando los datos urgentes a enviar son unos pocos bytes, que al ir solos resultaría una transmisión poco eficiente.

4.2.10 Opciones.

Este campo se utiliza únicamente al establecer una conexión. Indica el máximo tamaño de los segmentos que se puede recibir. Un segmento es cada una de las partes en las que se dividen los datos, por lo que el transmisor no debe enviar nunca paquetes más grandes de lo establecido en este campo de opción.

Si este campo no es usado el transmisor enviará paquetes de cualquier tamaño, ajustándose únicamente al campo ventana del receptor.

4.3 Estados de Conexión TCP

4.3.1 Establecimiento de Conexión.

Existen por lo menos dos paquetes encargados de establecer la conexión: uno que solicita el establecimiento de conexión, y otro que acepta dicha solicitud⁴

En realidad, el establecimiento de conexión TCP requiere aún otro parámetro, el cual da el nombre al establecimiento de la conexión como 3-way handshake (saludo de 3 pasos) el cual consiste en lo siguiente:

1. El cliente solicita al servidor una conexión
2. El servidor responde aceptando la conexión
3. El cliente responde aceptando la conexión

El tercer paso permite una sincronización exacta entre cliente y servidor, y además permite al cliente rechazar la conexión si no esta de acuerdo con la respuesta del servidor.

Los siguientes paquetes especiales utilizados en el 3-way handshake se caracterizan por sus flags:

- El cliente solicita conexión al servidor: flag **SYN**
- El servidor acepta la conexión: flags **SYN** y **ACK**
- El cliente acepta la conexión: flag **ACK**

⁴ HACK X CRACK No 21 (TCP- Comprendiendo los ataques de Internet)

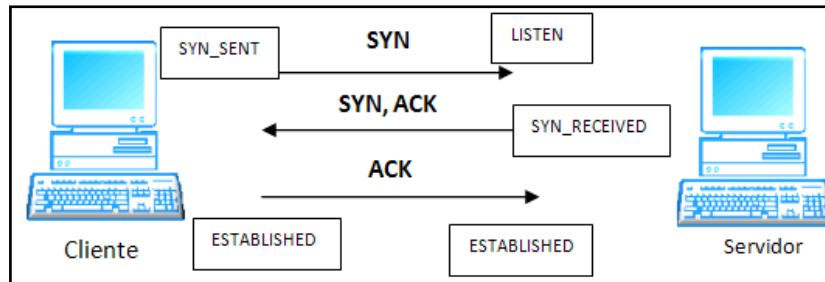


Figura. 2.6. Establecimiento de Conexión

Este siempre será el procedimiento a seguir, si alguno de esos flags no es enviado en el orden correcto, todo el establecimiento de conexión será anulado.

4.3.2 Cierre de Conexión.

Este constituye otro procedimiento que se puede llevar a cabo con cualquier conexión, y es el cierre de la misma.

En una conexión full dúplex es necesario que ambas partes se pongan de acuerdo sobre el momento en el que hay que cerrar la conexión.

En el caso de TCP, el mecanismo que se utiliza para conseguir esto, es que cada uno por su cuenta indique al otro el momento en que se cierra la conexión.

Una vez que la otra parte recibe el aviso de cierre de conexión, esperará a que el también de por culminada la conexión.

Para dar este tipo de avisos se envía un paquete especial que tiene activado un flag que sirve precisamente para indicar que se desea finalizar la conexión, este flag es el **FIN**.

A partir del momento que se envía un paquete con el flag **FIN**, ya no se envía ningún otro paquete (excepto en el caso que se tuviera que enviar un

paquete anterior al no recibir su confirmación de llegada), y solo se debe recibir los datos de la otra parte hasta que este envíe su paquete con el flag **FIN**.

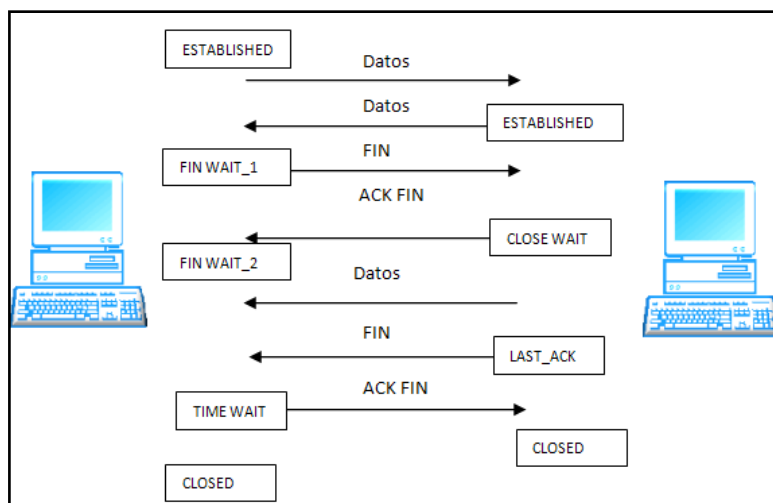


Figura. 2.7. Cierre de Conexión

4.3.3 Lista de estados TCP.

4.3.3.1 LISTEN.

Es el estado en el que permanece cualquier servidor en espera a que un cliente se conecte. Todos los puertos que se encuentren abiertos en el PC generará un socket TCP (o UDP, según el caso) en estado LISTEN. Cada vez que un cliente se conecte, se creará un socket en estado ESTABLISHED para ese cliente, y otro en estado LISTEN para esperar a otros clientes.

4.3.3.2 SYN_SENT.

Se entra en este estado cuando se solicita una conexión a un servidor, y aún no se ha recibido su aceptación o su rechazo.

4.3.3.3 SYN RECEIVED.

Se entra en este estado cuando tanto cliente como servidor han enviado sus correspondientes SYN.

4.3.3.4 ESTABLISHED.

Conexión establecida, es el estado habitual en el cual ya se ha aceptado la conexión por las dos partes involucradas.

4.3.3.5 FIN WAIT_1.

Se ha enviado un paquete FIN, se desea cerrar sesión pero aun se espera un paquete de confirmación de la otra parte, en este estado solo se puede recibir, pero no enviar.

4.3.3.6 FIN WAIT_2.

Se ha enviado un paquete FIN, y además se recibe la confirmación ACK del fin, por lo tanto la otra parte ya conoce la intención de cerrar la conexión.

4.3.3.7 CLOSE WAIT.

Se ha recibido el paquete FIN, sin embargo el host todavía tiene datos por enviar así que espera hasta transmitir todos los datos y enviar el FIN.

4.3.3.8 CLOSING.

Se espera a la última confirmación para cerrar definitivamente la conexión.

4.3.3.9 LAST ACK.

Se espera la confirmación (ACK) de FIN, cuando este host era el último que faltaba en enviar el FIN.

4.3.3.10 TIME_WAIT.

Se espera un tiempo prudencial hasta que el host que faltaba la ultima confirmación (ACK FIN) la reciba.

5. PROTOCOLO DE INTERNET (IP)

El protocolo IP sirve como protocolo universal para interconectar dos ordenadores en cualquier momento, lugar y tiempo. El protocolo de internet proporciona los medios necesarios para la transmisión de bloques de datos llamados datagramas desde el origen al destino, siendo el origen y el destino hosts identificados por direcciones de longitud fija. El protocolo de internet también se encarga, si es necesario, de la fragmentación y el ensamblaje de grandes datagramas para su transmisión a través de la red.

El Protocolo Internet está específicamente limitado a proporcionar las funciones necesarias para enviar un paquete de bits (datagrama) desde un origen a un destino a través de un sistema de redes interconectadas. No existen mecanismos para aumentar la fiabilidad de datos entre los extremos, control de flujo, secuenciamiento u otros servicios que se encuentran normalmente en otros protocolos host-a-host. El protocolo internet puede aprovecharse de los servicios de sus redes de soporte para proporcionar varios tipos y calidad de servicio.

5.1 Generalidades

Un módulo IP reside en cada host involucrado en la comunicación y en cada pasarela (Gateway) que interconecta redes. Estos módulos comparten reglas comunes para interpretar los campos de dirección y para fragmentar y ensamblar datagramas IP. Además, estos módulos (especialmente en las pasarelas) tienen procedimientos para tomar decisiones de encaminamiento y otras funciones.

El protocolo IP (Internet Protocol) implementa dos funciones básicas: direccionamiento y fragmentación.

La capa IP es la que se encarga de asignar direcciones únicas a cada máquina, para acceder a cada una de ellas como si de números de teléfono se tratase.

Los módulos internet usan las direcciones que se encuentran en la cabecera internet para transmitir los datagramas IP hacia sus destinos. La selección de un camino para la transmisión se llama encaminamiento.

El protocolo IP permite la fragmentación es decir divide los datagramas (paquetes IP) en segmentos lo suficientemente pequeños como para adaptarse a las capacidades tecnológicas de cada red. IP no solo se encarga de dividir los fragmentos, sino que además debe garantizar un mecanismo con el cual reconstruir los paquetes originales a partir de los fragmentos.

El protocolo IP utiliza cuatro mecanismos clave para implementar estos servicios: Tipo de Servicio (TOS), Tiempo de Vida (TTL), Opciones, y Suma de Control de Cabecera (checksum).

5.2 Formato de la Cabecera IP

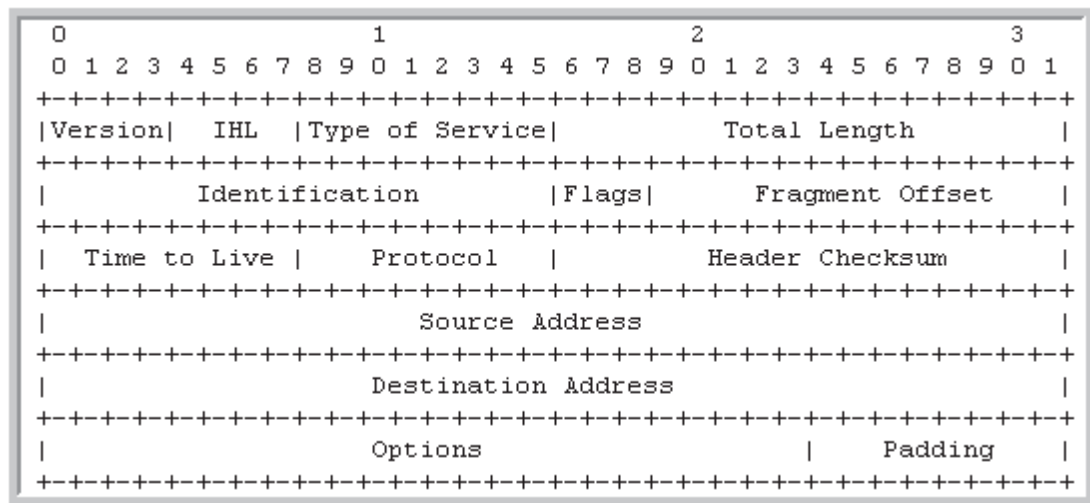


Figura. 2.8. Formato de la cabecera IP

5.2.1 Versión: 4bits.

Hace referencia a la versión IPv4 por tanto este campo valdrá siempre 4 (0100), sin embargo el próximo estándar será la versión IPv6 que definirá el funcionamiento de la nueva red Internet⁵

5.2.2 Longitud de la cabecera IP (IHL): 4bits.

Este campo indica la longitud de la cabecera IP medida en palabras de 32 bits. La cabecera IP puede contener opciones de tamaño variable, este campo nos indica en que punto termina la cabecera y comienza los datos del paquete.

⁵ Gabriel Verdejo Álvarez – “SEGURIDAD EN REDES IP: Los protocolos TCP/IP” ; HACK X CRACK No 25 (La capa IP – Encaminamiento de Paquetes)

Una cabecera sin ninguna opción mide 160 bits, es decir 5 filas de 32 bits, siendo este el valor mínimo para este campo y también el más típico, 5(0101).

5.2.3 Tipo de Servicio (TOS): 8bits.

El campo **TOS** consta de dos partes, la una de 3 bits especifica un nivel de prioridad, cuando mayor es el valor de estos 3 bits, mayor prioridad especificando en la siguiente tabla:

000	Datagramas rutinarios
001	Datagramas prioritarios
010	Datagramas de necesidad inmediata
011	Datagramas flash
100	Datagramas flash override
101	Datagramas críticos
110	Datagramas de control entre redes
111	Datagramas de control de red

Tabla.2.3. El Campo TOS

En caso de congestión en la red, los datagramas que serán descartados en primer lugar serán los que tengan prioridad 000.

La segunda parte del **TOS** son 5 bits que funcionan a modo de flags, cada bit es un indicador independiente de alguna característica del datagrama, estos son los significados de cada flag:

Bit 3	Requisitos de retardo (flag D)
Bit 4	Requisitos de ancho de banda (flag T)
Bit 5	Requisitos de fiabilidad (flag R)
Bit 6	Reservado para uso futuro
Bit 7	Reservado para uso futuro

Tabla. 2.4. Segunda Parte del TOS

Se toma en cuenta desde el bit 3, ya que los bits 0-2 especifican la prioridad (precedente), tal y como vemos en este esquema que resume toda la estructura del campo **TOS**:

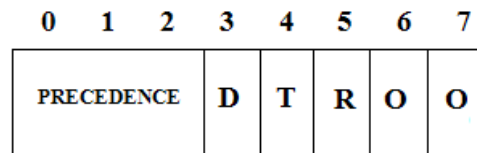


Figura. 2.9. Estructura del Campo TOS

Los bits 6 y 7 no se usan, por lo que se ponen siempre a 0. Los otros 3 flags (D, T, Y R) mantienen 3 características de la comunicación entre sí.

Si el **flag D (Delay)** está activo significa que el tipo de servicio que esta ofreciendo este datagrama requiere un retardo mínimo.

Si el **flag T (Throughput)** está activo significa que el tipo de servicio que está ofreciendo este datagrama requiere el máximo ancho de banda que se le pueda ofrecer por ejemplo para transmisiones de contenido multimedia.

Si el **flag R (Reliability)** esta activo significa que el tipo de servicio que está ofreciendo este datagrama requiere una máxima fiabilidad independiente de que la comunicación sea en tiempo real o del ancho de banda de la misma, cualquier aplicación que requiera precisión en los datos, como la transferencia de archivos binarios esta dentro de este tipo de servicio.

5.2.4 Longitud Total (Total Length): 16 bits.

Indica en bytes la longitud total de todo el datagrama, al ser 16 bits, el tamaño máximo de paquete será de 65536 bytes, lo cual está muy por encima del tamaño de paquete que pueda circular por cualquier red normal.

5.2.5 Identificación: 16 bits.

Es el campo que se utiliza para identificar el datagrama al que pertenece un fragmento, la combinación de este campo con la ip de origen, destino, y el numero de protocolo, identifica unívocamente un datagrama para permitir su ensamblaje a partir de sus fragmentos.

5.2.6 Indicadores (flags): 3bits

Bit 0	Reservado
Bit 1	Fragmentación no permitida (DF)
Bit 2	Faltan fragmentos (MF)

Tabla. 2.5. Indicadores (flags): 3 bits

El **bit 0** es de uso reservado, por lo que siempre se pondrá a cero

El flag **DF (Dont Fragment)**, en caso de estar activo, indica que este datagrama no puede ser fragmentado, por lo que en caso de que el datagrama sea demasiado grande para las capacidades tecnológicas de la red, tendrá que ser descartado.

El **flag MF (More Fragments)**, en caso de estar activo (1), indica que este fragmento no es el último de los que forman el datagrama original, en caso de estar en 0 indica que el fragmento es el último.

5.2.7 Desplazamiento del fragmento (Fragment Offset): 13 bits.

Indica la posición del fragmento dentro del datagrama original, en unidades de 8 octetos, es decir, 64 bits.

El primer fragmento tendrá siempre un desplazamiento 0. Si por ejemplo, el primer fragmento tenía un tamaño de 512 bytes, el segundo fragmento tendrá un desplazamiento de $512/8=64$

5.2.8 Tiempo de Vida (TTL: Time to Live): 8bits.

Indica el número máximo de Gateways que puede atravesar el datagrama antes de considerarse caduco y ser descartado.

Cada Gateway por el que pase el datagrama tendrá que restar 1 este valor, y si en algún caso llega a valor 0, el Gateway que dio este valor tiene que interrumpir en ese punto la transmisión del datagrama, y devolver a su transmisor original un mensaje **ICMP** de tipo **Time Exceeded**.

5.2.9 Protocolo: 8bits.

Este campo identifica el protocolo de nivel superior en la jerarquía, para que el módulo de la capa IP del receptor sepa como debe procesar el datagrama. Por ejemplo, el protocolo TCP se identifica con el valor 6 (00000110).

5.2.10 Suma de Comprobación de la cabecera (Header Checksum): 16 bits.

Los Gateways que procesan el datagrama a lo largo de todo el camino modifican la cabecera, es necesario recalcularse una nueva suma de comprobación en cada Gateway por el que pasa el datagrama.

Por tanto cada Gateway, comprueba el checksum de datagrama que ha recibido, lo recalcula y crea un nuevo checksum que sustituirá al anterior, este campo de la cabecera IP es modificado en cada Gateway por el que pasa el datagrama.

5.2.11 Dirección IP de origen (Source Address): 32 bits.

Una dirección IP ocupa 32 bits, este es uno de los campos más importantes de la cabecera IP, especifica la dirección IP del transmisor del datagrama. Es el campo a modificar cuando se hace IP spoofing.

5.2.12 Dirección IP de destino (Destination Address): 32 bits.

Constituye la dirección IP del receptor del datagrama.

5.2.13 Opciones (Options).

Este es uno de los campos más complejos de la cabecera IP, pues su longitud es variable.

Puede existir datagramas que no tengan ninguna opción y, por tanto, todo este campo sea inexistente, los datagramas sin opciones son los más

comunes pero también puede haber datagramas con un gran número de opciones, y cada uno de ellos además con una longitud variable.

El campo Options consta de una, ninguna, o varias opciones, siendo el formato de cada una de ellas variable. El único campo que tienen en común todas las opciones es un campo de un byte que sirve para identificar el tipo de opción.

Este byte llamado Tipo de opción (**opción-type**), consta a su vez de varios campos que, combinados, identifican un tipo concreto de opción.

Esta es la estructura del **option-type**:

Bit 0	Bit1 y 2	Bits 3 a 7
Copied flag	Option class	Option number

Tabla. 2.6. Estructura del Option-type

El primer bit es el bit de copiado (**Copied flag**). Este bit se utiliza sólo en datagramas fragmentados. En caso de que este bit este a 1 significa que esta opción se repite en todos los fragmentos del mismo datagrama, es decir, es una opción del datagrama original, y no solo del fragmento.

A continuación tenemos los 2 bits de clase de opción (**Option class**). Existen 4 clases definidas, de las cuales solo se utilizan 2:

00	Clase de control
01	Reservada para uso futuro
10	Clase de depuración
11	Reservado para uso futuro

Tabla. 2.7. Option class

Por ultimo, tenemos 5 bits que especifican el tipo concreto de opción dentro de esa clase (**option number**). Esta es la lista de opciones definidas para cada clase:

Clase	Option	Longitud	Descripción
00	0	-	Fin de lista de opciones
00	1	-	Sin operación
00	2	11	Opciones de seguridad
00	4	Variable	Encaminamiento debilmente especificado por el transmisor
00	9	Variable	Encaminamiento estrictamente especificado por el transmisor
00	7	Variable	Registro de ruta
00	8	4	Identificador de flujo
10	4	variable	Sello de tiempo

Tabla. 2.8. Option Number

5.2.14 Padding.

Como se mencionó anteriormente, el campo IHL mide la longitud de la cabecera en palabras de 32 bits. Esto no significa que las opciones de la cabecera tengan que ajustarse necesariamente a ese tamaño. Por este motivo, en caso de que el tamaño de las opciones no sea multiplo de 32 bits, será necesario rellenar con ceros los bits hasta que se complete una fila de 32 bits. Estos ceros son los que aparecen como Padding en la cabecera del datagrama IP.

Existe un innumerable de cosas que se podría decir sobre la capa IP, sobre todo desde el punto de vista práctico, sobre todo el como utilizar esta

información para comprometer la seguridad de un sistema, y como configurar un sistema para evitar estos problemas.

5.3 Principio de Funcionamiento del Protocolo IP

5.3.1 Diferencia entre Paquete, Segmento y Datagrama.

TCP produce un bloque de datos que se denomina segmento para ser transmitido por IP. UDP produce un datagrama para ser transmitido por IP. Entonces IP crea sus propios datagramas a partir de lo que recibe de TCP o de UDP. Si el segmento de TCP o el datagrama de UDP mas los encabezados de IP es de tamaño suficientemente pequeño para enviarlo en un único paquete a través del cable, IP crea un paquete, que en si es lo mismo que un datagrama IP. Si el datagrama de IP es demasiado extenso para el cable, esto es, si sobrepasa la unidad máxima de transmisión (MTU) del medio, entonces IP fragmenta el datagrama en paquetes más pequeños, adecuados para la MTU del medio⁶

El modelo de operación para transmitir un datagrama de una aplicación a otra se ilustra en el siguiente escenario:

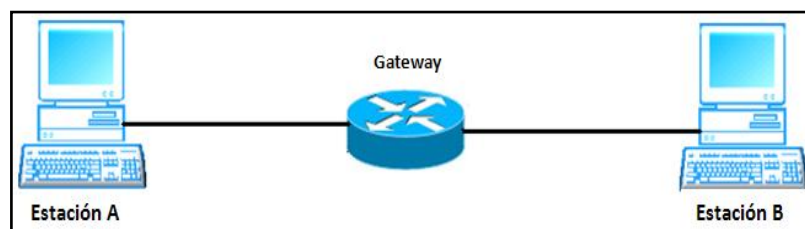


Figura. 2.10. Modelo de Operación

⁶ HACK X CRACK No 25 (La capa IP – Encaminamiento de Paquetes)

Tenemos dos maquinas A y B que se comunican entre sí a través de un Gateway, en un caso real lo normal es que no exista un único Gateway, si no todo un conjunto de ellos para comunicar A con B.

Suponiendo que A quiere enviar un datagrama a B, y que ese paquete forma parte de la comunicación entre dos aplicaciones que corren en A y B, la aplicación remitente prepara el contenido a transmitir y llama a su capa IP local (sistema operativo) para enviar esos datos como un datagrama, proporciona la dirección de destino y además incluye en estos datos las cabeceras de protocolos de nivel superior como podría ser TCP y otros parámetros como argumentos de la llamada, como ejemplo de aplicaciones que pueden estar en las estaciones de trabajo tenemos un servidor web corriendo en B, y un navegador (Opera, Firefox, Internet Explorer....) corriendo en A.

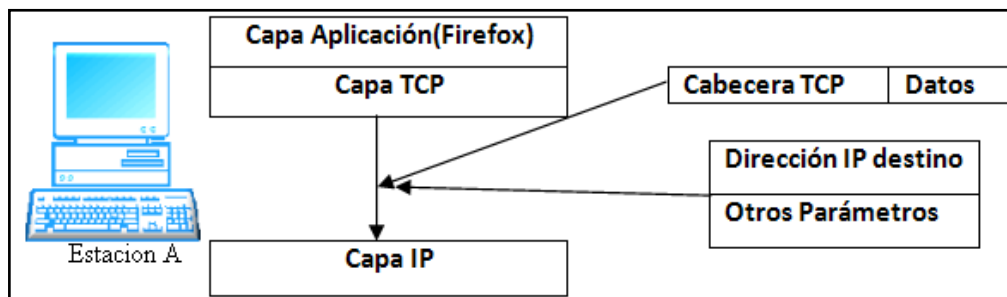


Figura. 2.11. Modelo de Operación para transmitir un Datagrama de una Aplicación a otra

Cuando la capa IP tiene el paquete y los parámetros necesarios, tendrá que construir su propia cabecera, formando así un datagrama que dependiendo del tamaño de este podría ser fragmentado por ser demasiado grande, una vez formado el datagrama, este será enviado con una serie de parámetros a la capa inferior conocida como capa de acceso o de enlace.

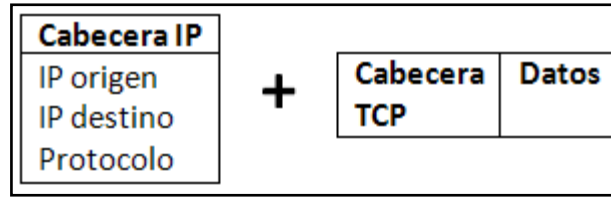


Figura. 2.12. Datagrama Formado

El parámetro más importante es la dirección del Gateway que es la dirección que pasa de la capa IP a la capa de Acceso y que no constituye la dirección de destino B.

El Gateway es la única máquina que establece conexión directa con A y será el encargado de enviar el datagrama hacia B.

Una vez el datagrama ya está en la capa de acceso, junto con los parámetros necesarios, esta capa se encargará de añadir al datagrama su propia cabecera para luego enviar el paquete a través del medio físico.

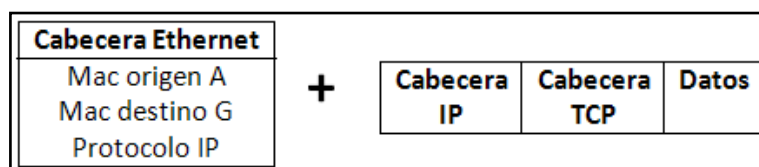


Figura. 2.13. Paquete Enviado

El paquete llegará a cada una de las máquinas conectadas a la red local incluyendo al Gateway, que al comprobar la dirección de destino será la única que se quede con el paquete, sin embargo dentro de la red puede existir alguna máquina en modo promiscuo usando un sniffer para capturar todo el tráfico de la red, aunque no vaya dirigido a ella.

Cuando el paquete llega al Gateway, como ya se utilizó la información de la cabecera de Acceso, para saber que el paquete va dirigido a él, se puede prescindir de esta cabecera, y obtener el datagrama tal y como lo genero la capa IP de la máquina A, con el fin de descubrir que la IP de destino no es la

suya si no la de la máquina B y retransmite el paquete para que pueda llegar a su destino.

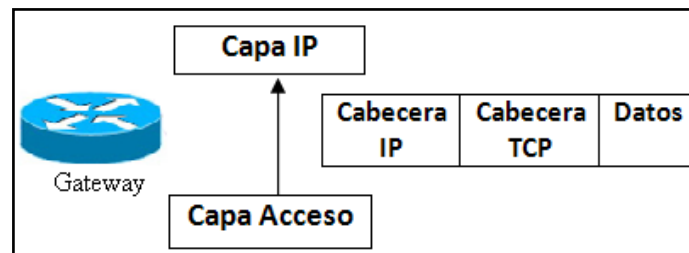


Figura. 2.14. Datagrama como lo Genera la Capa IP

El Gateway genera una nueva cabecera de enlace que añadirá al datagrama para transmitirlo a través del medio físico que une al Gateway con B.

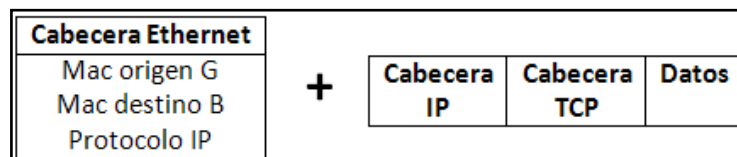


Figura. 2.15. Datagrama para trasmitirlo

El paquete llegará a todas las máquinas conectadas a la red local incluyendo B que gracias a la nueva cabecera generada por el Gateway sabrá que el paquete va destinado a él. Al no necesitar más la cabecera de Acceso la destruye y pasa al resto del paquete (datagrama), la capa IP de la máquina B determina que el datagrama va dirigido a una aplicación en este host ya que la dirección IP de destino coincide con su dirección IP, y busca en la cabecera IP el dato con el cual pasar a la capa superior del datagrama.

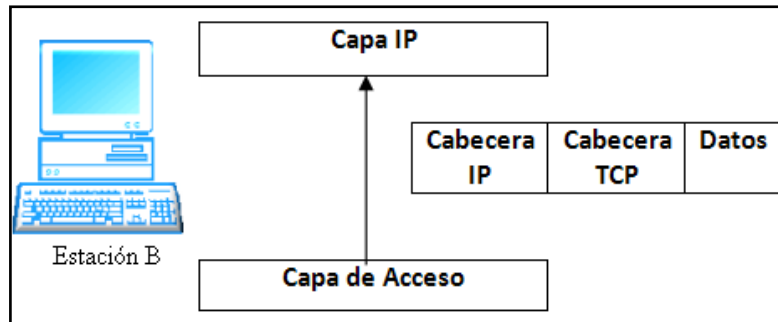


Figura. 2.16. Modelo de Operación para transmitir un Datagrama a la red local B

Y luego solo tendrá que enviar el datagrama a las capas superiores donde la aplicación de B como ejemplo un servidor web (Apache) procesará los datos provenientes de A.

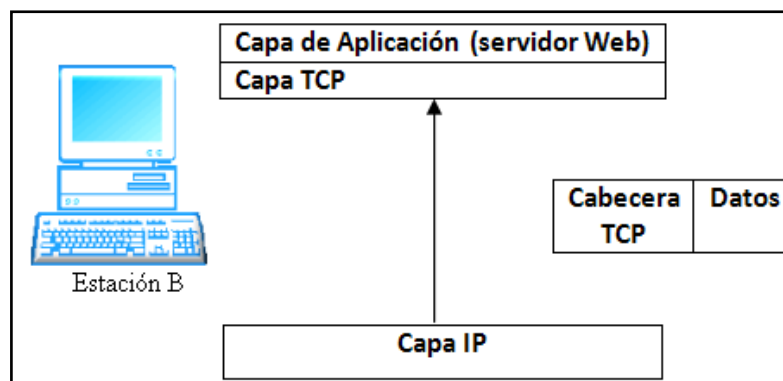


Figura. 2.17. Modelo de Operación para transmitir un Datagrama a una capa de Aplicación

5.4 Descripción de Funciones del Protocolo IP

La función o propósito del protocolo de Internet (IP) es mover datagramas a través de un conjunto de redes interconectadas basándose en la interpretación de una dirección IP. Esto se consigue pasando los datagramas desde una capa IP a otra hasta que se alcanza el destino. Las capas de internet residen en hosts y encaminadores (gateways).

En el enrutamiento de mensajes desde una capa IP a otra, los datagramas pueden necesitar atravesar una red cuyo tamaño máximo de paquete es menor que el tamaño del datagrama. Para superar esta dificultad se proporciona un mecanismo de fragmentación en el protocolo IP.

5.4.1 Direccionamiento.

Se establece una distinción entre nombres, direcciones y rutas.

- Un nombre indica que buscamos.
- Una dirección indica donde está lo que buscamos.
- Una ruta indica cómo llegar a esa dirección.

Es tarea de los protocolos de nivel superior a IP convertir nombres en direcciones, el protocolo **DNS** es el que traduce nombres a direcciones IPs y viceversa.

IP también se aprovecha de funciones llevadas a cabo por protocolos inferiores, es decir por la capa de Acceso.

La capa de Acceso es la que se encarga de mantener la traducción de direcciones IP a direcciones físicas (MAC). Si se utiliza Ethernet en la capa de Acceso, se necesitará del protocolo **ARP** (Address Resolution Protocol) para mantener la traducción entre direcciones IP y direcciones de red local (MAC).

Las direcciones IP son de una longitud fija de 4 octetos (32 bits). Una dirección comienza por un número de red, seguido de la dirección local (llamada campo). Hay 3 formatos o clases de direcciones internet:

- En la Clase A, el bit más significativo es "0", los 7 bits siguientes son de red, y los 24 bits restantes son la dirección local.

- En la Clase B, los dos bits más significativos son uno-cero ("10"), los 14 bits siguientes son de red y los últimos 16 bits son la dirección local.
- En la Clase C, los tres bits más significativos son uno-uno-cero ("110"), los 21 bits siguientes son de red y los 8 restantes son la dirección local.

5.4.2 Fragmentación.

5.4.2.1 Tráfico Fragmentado.

Los ataques de disponibilidad y de denegación de servicio utilizan tráfico altamente fragmentado para agotar los recursos del sistema. Se debe analizar el tráfico fragmentado y descubrir si es una fragmentación normal o se trata de algún tipo de ataque por fragmentación.

5.4.2.2 Teoría de la fragmentación.

Según la tecnología utilizada en cada red, los paquetes tendrán limitado su tamaño máximo, al encargarse la capa IP de esta función, la fragmentación tiene que ser transparente para los protocolos superiores, como TCP, o los protocolos de aplicación⁷

La fragmentación se produce cuando un datagrama IP que viaja por una red tiene que atravesar una red con unidad de transmisión máxima (MTU) menor que el tamaño del datagrama. La MTU o tamaño máximo de un datagrama IP para Ethernet es de 1500 bytes. Si un datagrama es mayor necesita ser fragmentado por medio de un enrutador que se dirija a la red Ethernet, así los fragmentos continúan hacia su destino, donde el host de destino los reconstruye.

⁷ Joaquín García Alfaro/ Xavier Perramón Tornil (Aspectos Avanzados de Seguridad en Redes)

Dentro del datagrama IP existen campos los cuales indican que un datagrama pertenece a ese fragmento, y otro campo que indica que posición ocupa ese fragmento dentro del datagrama original.

Para llevar a cabo la identificación del datagrama al que pertenece el fragmento se recurre a una combinación de campos, de los cuales el más importante es el llamado **Identification**.

Este campo es un número que puede identificar unívocamente al datagrama, y que junto con las IPs de origen, destino, y el protocolo de nivel superior (TCP, UDP) son siempre únicos para cada datagrama.

Para saber qué posición ocupa el fragmento dentro de ese datagrama se recurre a otro campo, llamado **Fragment Offset**, que indica en que byte del datagrama original comienza ese fragmento.

Un datagrama IP puede ser marcado como **dont Fragment**. Todo datagrama IP así marcado no será fragmentado entre distintas redes bajo ninguna circunstancia. Si un datagrama IP marcado así no puede ser entregado en su destino sin fragmentarlo, entonces debe ser descartado.

El indicador **More-Fragment** indica (puesto a cero) el último fragmento. Estos campos proporcionan información suficiente para ensamblar luego los datagramas.

➤ **Ejemplo de Fragmentación**

Ethernet será la capa de enlace para este ejemplo (MTU 1500). La figura 1.16 representa un datagrama no fragmentado, el cual tiene un encabezado IP, que normalmente es de 20 bytes, pudiendo ser mayor si se incluye las opciones IP.

Cabecera IP	Bytes de datos encapsulados
20 bytes	1480 bytes

Figura. 2.18. Datagrama Ethernet (MTU=1500)

Los routers utilizan la información del encabezamiento IP para dirigir el datagrama a su destino, después del encabezamiento IP, se encapsula alguna clase de datos pudiendo ser estos un protocolo como TCP, UDP o ICMP.

La figura 1.17 representa un datagrama de 4028 bytes, con petición eco ICMP utilizado para ver como se produce la fragmentación, por lo tanto el datagrama de 4028 bytes tendrá que dividirse en fragmentos de 1500 bytes para atravesar la red Ethernet. Cada uno de estos paquetes fragmentados de 1500bytes o menos tendrá un encabezamiento IP de 20 bytes como fragmento inicial, quedando un máximo de 1480 bytes para datos en cada fragmento.

Cabecera IP	Cabecera ICMP	Datos ICMP	Datos ICMP	Datos ICMP
20 bytes	8 bytes	1500 bytes	1500 bytes	1068 bytes

Figura. 2.19. Fragmento de 4028 bytes dividido en dos fragmentos de 1500 bytes y uno de 1068 bytes.

➤ **Fragmento inicial.** El fragmento IP original se clonará para que contenga números de identificación de fragmento (Identification) idénticos para el primer y los restantes fragmentos.

El primer fragmento es el único que lleva consigo el encabezamiento de mensaje ICMP, este encabezamiento no se clona en los fragmentos asociados posteriores, este hecho hace del primer fragmento significativo al ser este el que identifica el tipo de protocolo y el tipo de petición ICMP.

El primer fragmento tiene un valor de desplazamiento igual a 0 (offset=0), una longitud de 1480 bytes, 1472 bytes de datos y 8 de encabezamiento ICMP, y como existen mas fragmentos, se establece el indicador de mas fragmentos (MF=1), para indicar que hay mas fragmentos a continuación.

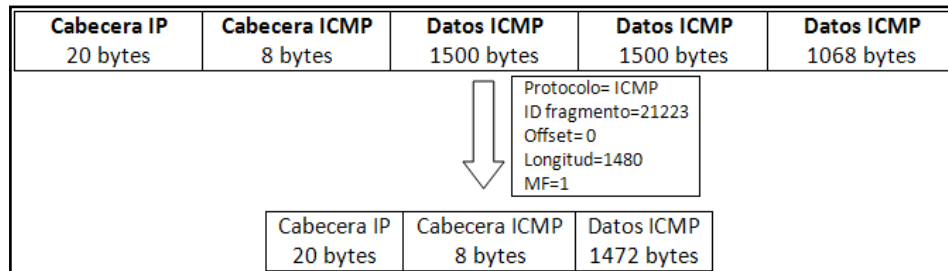


Figura. 2.20. Fragmento inicial

Además de los campos principales de la cabecera IP, como Ip de origen, destino y protocolo (ICMP), tenemos los campos específicos para la fragmentación. El ID de fragmento con un valor de 21223 es el identificador común para todos los fragmentos.

➤ **Segundo Fragmento.** En el siguiente fragmento, figura 1.18 se clona un encabezamiento IP del original con un número de identificación de fragmento idéntico, y una longitud de 1480 bytes y como le sigue un fragmento mas se establece el indicador (MF=1).

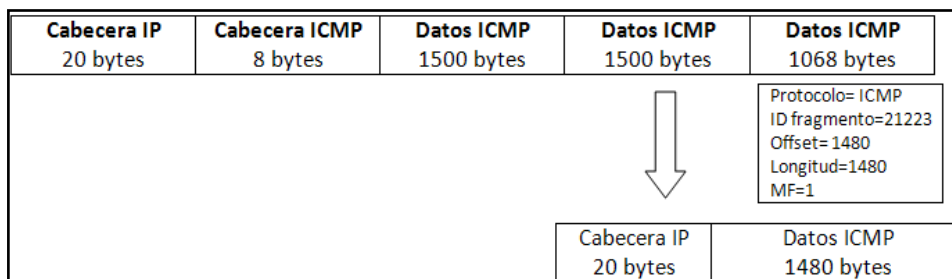


Figura. 2.21. Segundo Fragmento

Este segundo fragmento también tiene una longitud de 1480 bytes (datos ICMP), el encabezamiento ICMP del primer fragmento no se clona junto con los datos ICMP, esto significa que mediante este fragmento no se puede identificar el tipo de mensaje ICMP en este caso, una petición eco ICMP.

➤ Último fragmento

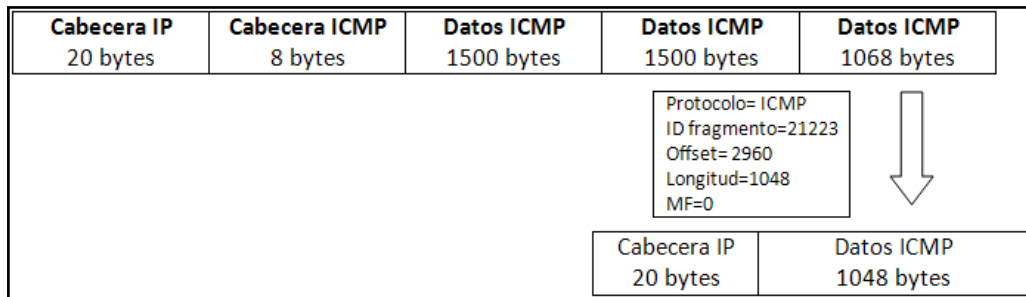


Figura. 2.22. Último Fragmento

En el último fragmento se reservan 20 bytes para encabezamiento IP. El resto de bytes de datos ICMP están en los datos de este fragmento. El ID de fragmento es 21223, y no está establecido el indicador de mas fragmentos (MF=0), porque este es el último fragmento. El valor de desplazamiento es de 2960 (suma de los dos fragmentos anteriores de 1480 bytes). Solo hay 1048 bytes de datos (el resto del mensaje ICMP). Este fragmento al igual que el segundo no tiene encabezado ICMP por tanto no se puede identificar que se trata de una petición ICMP.

6. PROTOCOLO ICMP

El protocolo **ICMP** es el encargado de realizar el control de flujo de los datagramas IP que circulan por una red TCP/IP. Este protocolo consta de varias funcionalidades que permiten realizar desde la comunicación de situaciones con anomalías (por ejemplo, indicar que no se ha podido realizar la

entrega de un datagrama IP) hasta la comprobación del estado de una máquina en la red⁸

El protocolo ICMP únicamente informa de incidencias en la red pero no toma ninguna decisión. Esto será responsabilidad de las capas superiores.

Aunque el objetivo original del protocolo ICMP es el de notificar errores y condiciones inusuales (que requieren una especial atención respecto al protocolo IP), es posible poder realizar un uso indebido de este protocolo para obtener la información y el estado de un sistema remoto.

Los mensajes ICMP comienzan con un campo de 8 bits que contiene el tipo de mensaje, según se muestra en la tabla siguiente. El resto de campos son distintos para cada tipo de mensaje ICMP.

Campo de Tipo	Tipo de mensaje ICMP
0	Respuesta de eco (Echo Replay)
3	Destino inaccesible (Destination Unreachable)
4	Disminución de tráfico desde el origen (Source Quench)
5	Redireccionar, cambio de ruta (Redirect)
8	Solicitud de eco (Echo)
11	Tiempo excedido para un datagrama (Time Exceeded)
12	Problema de parámetros (Parameter Problem)
13	Solicitud de marca de tiempo (Timestamp)
14	Respuesta de marca de tiempo (Timestamp Replay)
15	Solicitud de información (Information Request)
16	Respuesta de información (Information Reply)
17	Solicitud de máscara (address mask)
18	Respuesta de máscara (address mask reply)

Tabla. 2.9. Protocolo ICMP

⁸ Fuente: Redes de Información (RIN) - Universidad Tecnológica Nacional – F.R.C

6.1 Solicitud y respuesta de eco

Los mensajes de solicitud y respuesta de eco, tipos 8 y 0 respectivamente, se utilizan para comprobar si existe comunicación entre 2 hosts a nivel de la capa de red. Sin embargo, no dicen nada de las capas de transporte y de aplicación las cuales podrían estar mal configuradas; por ejemplo, la recepción de mensajes de correo electrónico puede fallar aunque exista comunicación IP con el servidor de correo.

La orden **PING** envía mensajes de solicitud de eco a un host remoto e informa de las respuestas. Veamos su funcionamiento, en caso de no producirse incidencias en el camino.

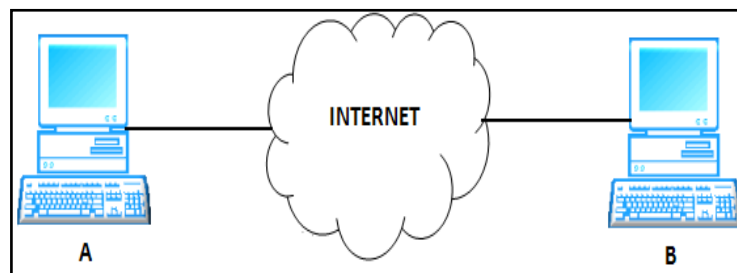


Figura. 2.23. Orden PING

1. A envía un mensaje ICMP de tipo 8 (*Echo*) a B
2. B recibe el mensaje y devuelve un mensaje ICMP de tipo 0 (*Echo Reply*) a A
3. A recibe el mensaje ICMP de B y muestra el resultado en pantalla

```
A>ping www.espe.edu.ec -n 1
```

Haciendo ping a 192.188.58.167 con 32 bytes de datos:

Respuesta desde 192.188.58.167: bytes=32 tiempo<1ms TTL=253

En la orden anterior se utiliza el parámetro "-n 1" para que el host A únicamente envíe 1 mensaje de solicitud de eco. Si no se especifica este parámetro se enviarían 4 mensajes (y se recibirían 4 respuestas).

Si el host de destino no existiese o no estuviera correctamente configurado recibiríamos un mensaje ICMP de tipo 11 (*Time Exceeded*).

```
A>ping www.espe.edu.ec -n 1
```

Haciendo ping a 192.188.58.167 con 32 bytes de datos:
Tiempo de espera agotado.

Si tratamos de acceder a un host de una red distinta a la nuestra y no existe un camino para llegar hasta él, es decir, los routers no están correctamente configurados o estamos intentando acceder a una red aislada o inexistente, recibiríamos un mensaje ICMP de tipo 3 (*Destination Unreachable*).

```
A>ping 1.1.1.1 -n 1
```

Haciendo ping a 1.1.1.1 con 32 bytes de datos:
Respuesta desde 192.168.0.1: Host de destino inaccesible.

ICMP es un protocolo único porque no utiliza puertos para comunicarse, como lo hacen los protocolos de transporte. Los mensajes ICMP se pueden perder y no ser entregados, nunca se utiliza para enviar un error de otros mensajes ICMP, además, se puede transmitir ICMP a muchos hosts, porque no hay ninguna conexión exclusiva.

6.2 Mensajes ICMP de tiempo excedido.

El campo TTL (tiempo de vida) de los datagramas IP impiden que un mensaje esté dando vueltas indefinidamente por la red. El número contenido en este campo disminuye en una unidad cada vez que el datagrama atraviesa

un router. Cuando el TTL de un datagrama llega a 0, éste se descarta y se envía un mensaje ICMP de tipo 11 (*Time Exceeded*) para informar al origen.

Los mensajes ICMP de tipo 11 se pueden utilizar para hacer una traza del camino que siguen los datagramas hasta llegar a su destino. ¿Cómo? Enviando una secuencia de datagramas con TTL=1, TTL=2, TTL=3, TTL=4, etc... hasta alcanzar el host o superar el límite de saltos (30 si no se indica lo contrario). El primer datagrama caducará al atravesar el primer router y se devolverá un mensaje ICMP de tipo 11 informando al origen del router que descartó el datagrama. El segundo datagrama hará lo propio con el segundo router y así sucesivamente. Los mensajes ICMP recibidos permiten definir la traza.

La orden **TRACERT** (traceroute en entornos Unix) hace una traza a un determinado host. TRACERT funciona enviando mensajes ICMP de solicitud de eco con distintos TTL; traceroute, en cambio, envía mensajes UDP. Si la comunicación extremo a extremo no es posible, la traza nos indicará en qué punto se ha producido la incidencia. Existen algunas utilidades en Internet, como Visual Route, que conocen la localización geográfica de los principales routers de Internet. Esto permite dibujar en un mapamundi el recorrido que siguen los datagramas hasta llegar a un host.

A>tracert 130.206.1.2

Traza a la dirección sun.rediris.es [130.206.1.2]

sobre un máximo de 30 saltos:

1 1 ms 1 ms 1 ms PROXY [192.168.0.1]

2 122 ms 118 ms 128 ms MADR-X27.red.retevision.es [62.81.1.102]

3 143 ms 232 ms 147 ms MADR-R2.red.retevision.es [62.81.1.92]

4 130 ms 124 ms 246 ms MADR-R16.red.retevision.es [62.81.3.8]

5 590 ms 589 ms 431 ms MADR-R12.red.retevision.es [62.81.4.101]

6 612 ms 640 ms 124 ms MADR-R10.red.retevision.es [62.81.8.130]

7 259 ms 242 ms 309 ms 193.149.1.28

8 627 ms 752 ms 643 ms 213.0.251.42

9 137 ms 117 ms 118 ms 213.0.251.142
10 109 ms 105 ms 110 ms A1-2-1.EB-Madrid00.red.rediris.es
[130.206.224.81]
11 137 ms 119 ms 122 ms A0-0-0-1.EB-Madrid3.red.rediris.es
[130.206.224.86]
12 109 ms 135 ms 115 ms sun.rediris.es [130.206.1.2]
Traza completa.

7. SEGURIDAD TCP/IP

En cada capa del modelo TCP/IP pueden existir distintas vulnerabilidades y un atacante puede explotar los protocolos asociados a cada una de ellas⁹

Cada día se descubren nuevas deficiencias, la mayoría de las cuales se hacen públicas por organismos internacionales, tratando de documentar, si es posible, la forma de solucionar y contrarrestar los problemas. A continuación se presenta algunas de las vulnerabilidades más comunes de las distintas capas del modelo TCP/IP

7.1 Vulnerabilidades de la capa de Acceso.

Las vulnerabilidades de la capa de Acceso están estrechamente ligadas al medio sobre el que se realiza la conexión. Esta capa presenta problemas de control de acceso y de confidencialidad.

Son ejemplos de vulnerabilidades a este nivel los ataques a las líneas punto a punto: desvío de los cables de conexión hacia otros sistemas,

⁹ Joaquín García Alfaro/ Xavier Perramón Tornil (Aspectos Avanzados de Seguridad en Redes)

interceptación intrusiva de las comunicaciones (pinchar la línea), escuchas no intrusivas en medios de transmisión sin cables, etc.

7.2 Vulnerabilidades de la capa de Red.

En esta capa se puede realizar cualquier ataque que afecte un datagrama IP. Se incluyen como ataques contra esta capa las técnicas de *sniffing*, la suplantación de mensajes, la modificación de datos, los retrasos y denegación de servicio.

Cualquier atacante puede suplantar un paquete si indica que proviene de otro sistema. La suplantación de un mensaje se puede realizar, por ejemplo: En esta capa, la autenticación de los paquetes se realiza a nivel de máquina (por dirección IP) y no a nivel de usuario. Si un sistema suministra una dirección de máquina errónea, el receptor no detectara la suplantación. Para conseguir su objetivo, este tipo de ataques suele utilizar otras técnicas, como la predicción de números de secuencia TCP, el envenenamiento de tablas caché, etc.

Por otro lado, los paquetes se pueden manipular si se modifican sus datos y se reconstruyen de forma adecuada los controles de las cabeceras. Si esto es posible, el receptor será incapaz de detectar el cambio.

7.3 Vulnerabilidades de la capa de Transporte.

La capa de transporte transmite información TCP o UDP sobre datagramas IP. En esta capa podemos encontrar problemas de autenticación, de integridad y de confidencialidad. Algunos de los ataques más conocidos en esta capa son las denegaciones de servicio debidas a protocolos de transporte.

En cuanto a los mecanismos de seguridad incorporados en el diseño del protocolo TCP (como las negociaciones involucradas en el establecimiento de una sesión TCP), existe una serie de ataques que aprovechan ciertas deficiencias en su diseño. Una de las vulnerabilidades más graves contra estos mecanismos de control puede ser la posibilidad de interceptación de sesiones TCP establecidas, con el objetivo de secuestrarlas y dirigir las a otros equipos con fines deshonestos.

Estos ataques de secuestro se aprovechan de la poca exigencia en el protocolo de intercambio de TCP respecto a la autenticación de los equipos involucrados en una sesión. Así, si un usuario hostil puede observar los intercambios de información utilizados durante el inicio de la sesión y es capaz de interceptar con éxito una conexión en marcha con todos los parámetros de autenticación configurados adecuadamente, podría secuestrar la sesión.

7.4 Vulnerabilidades de la Capa de Aplicación.

Como en el resto de niveles, la capa de aplicación presenta varias deficiencias de seguridad asociadas a sus protocolos. Algunos ejemplos de deficiencias de seguridad a este nivel podrían ser los siguientes:

7.4.1 Servicio de nombres de dominio (DNS).

Normalmente, cuando un sistema solicita conexión a un servicio, pide la dirección IP de un nombre de dominio y envía un paquete UDP a un servidor DNS; entonces, este responde con la dirección IP del dominio solicitado o una referencia que apunta a otro DNS que pueda suministrar la dirección IP solicitada.

Un servidor DNS debe entregar la dirección IP correcta pero, además, también puede entregar un nombre de dominio, dada una dirección IP u otro tipo de información.

En el fondo, un servidor de DNS es una base de datos accesible desde internet. Por lo tanto, un atacante puede modificar la información que suministra esta base de datos o acceder a información sensible almacenada en la base de datos por error, pudiendo obtener información relativa a la topología de la red de una organización.

7.4.2 Telnet.

Normalmente, el servicio Telnet autentica al usuario mediante la solicitud del identificador de usuario y su contraseña, que se transmiten en texto plano por la red.

Así, al igual que el resto de servicios de internet que no protegen los datos mediante mecanismos de protección, el protocolo de aplicación Telnet hace posible la captura de aplicación sensible mediante el uso de técnicas de *sniffing*.

Actualmente existen otros protocolos a nivel de aplicación (como, por ejemplo, SSH) para acceder a un servicio equivalente a Telnet pero de manera segura (mediante autenticación fuerte). Aun así, el hecho de cifrar el identificador del usuario y la contraseña no impide que un atacante que las conozca acceda al servicio.

7.4.3 File Transfer Protocol (FTP).

Al igual que Telnet, FTP es un protocolo que envía la información sin ningún tipo de encriptación (tanto por el canal de datos como por el canal de

comandos). Al enviar el identificador de usuario y la contraseña por una red potencialmente hostil, presenta las mismas deficiencias de seguridad que el protocolo Telnet.

FTP permite la conexión anónima a una zona restringida en la cual sólo se permite la descarga de archivos. De este modo, se restringen considerablemente los posibles problemas de seguridad relacionados con la captura de contraseñas, sin limitar sus principales funcionalidades.

7.4.4 Hypertext Transfer Protocol (HTTP).

El protocolo HTTP es el responsable del servicio *World Wide Web*. Una de sus vulnerabilidades más conocidas procede de la posibilidad de entrega de información por parte de los usuarios del servicio. Esta entrega de información desde el cliente de HTTP es posible mediante la ejecución remota de código en la parte del servidor.

La ejecución de este código por parte del servidor suele utilizarse para dar el formato adecuado tanto a la información entregada por el usuario como a los resultados devueltos (para que el navegador del cliente la pueda visualizar correctamente). Si este código que se ejecuta presenta deficiencias de programación, la seguridad del equipo en el que esté funcionando el servidor podría estar en peligro.

CAPITULO 3

ANALISIS DEL TRÁFICO TCP/IP DE LA RED DE PRUEBAS DEL DEPARTAMENTO DE ELECTRICA Y ELECTRÓNICA DE LA ESPE (DEEE)

1. INTRODUCCIÓN

En este capítulo se estudia los parámetros que rigen el comportamiento del tráfico TCP/IP, de la red del departamento de eléctrica y electrónica de la ESPE (DEEE), el cual posee una conexión permanente a Internet.

Se realiza un análisis detallado de todo el tráfico de la RED, comentando los distintos resultados obtenidos y desglosándolos por día, además de un estudio consistente de la monitorización del tráfico que circuló por cada una de las interfaces que forman parte del servidor. El análisis busca determinar el tipo de protocolo que mayor tráfico generó, cantidad y tamaño de paquetes de acuerdo a cada protocolo, entre otras características que determinen patrones propios del tráfico TCP/IP, para posteriormente en el capítulo 4 obtener un modelo que represente el comportamiento normal del tráfico TCP/IP que circula por la red de pruebas.

Para cumplir con el objetivo se utilizó la ayuda del software FLUKE NETWORK PROTOCOL EXPERT, herramienta que permite monitorizar el tráfico que circula por la red, y un ordenador con sistema operativo LINUX (Centos 5.9) para reproducir el tráfico capturado y simular el tráfico que se transmite en tiempo real.

2. GENERALIDADES DE LA RED DE ESTUDIO

2.1 Arquitectura de la Red

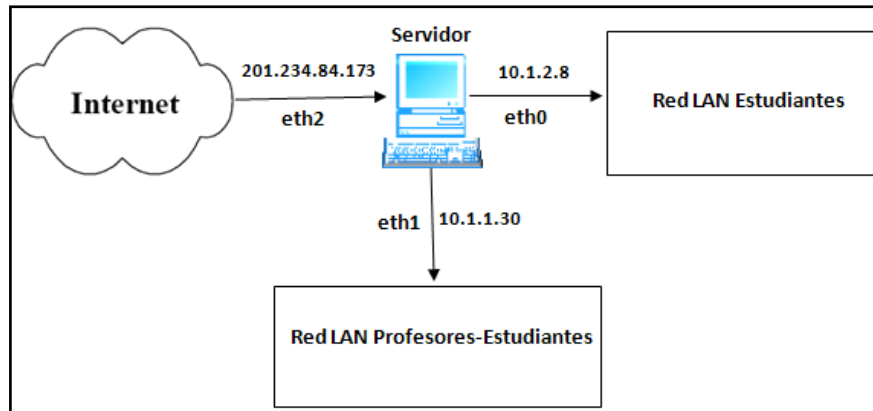


Figura. 3.1. Red del Departamento de Eléctrica y Electrónica de la ESPE (DEEE)

La red del departamento de Eléctrica y Electrónica de la ESPE (DEEE) está conformada por un servidor con acceso a internet a través de una dirección pública (201.234.84.173), por este servidor circula todo el tráfico dirigido a la Red LAN de estudiantes y Red LAN de Profesores – Estudiantes

Las aplicaciones que usa la red incluyen: transferencias de archivos a los sistemas de almacenamiento local, programas especializados, correo electrónico y transferencia de archivos dentro de la red y fuera de ella.

El objetivo de la red propuesta es el de permitir un análisis completo y exhaustivo del tráfico que circula por ella. Es por esto que existen diferentes aspectos estructurales que se debe tener en cuenta

2.1.1 Aspectos Estructurales

2.1.1.1 Conexión a Internet

Uno de los objetivos es el de monitorizar el acceso a Internet, de esta forma las características de la conexión (ubicación, acceso administrativo, ancho de banda...) determinarán la posibilidad o no de realizar esta tarea. La monitorización en tiempo real es un tema difícil y que no siempre es posible.

Por otro lado, la monitorización de un sistema conectado a Internet requiere de privilegios de administrador en los sistemas de comunicaciones (*routers*) y los servidores, lo que limita las posibilidades a sistemas controlados por nosotros mismos.

2.1.1.2 Recursos disponibles

La disponibilidad de recursos para la realización de este trabajo no es infinita, lo que nos obliga a utilizar únicamente aquellas herramientas disponibles.

La imposibilidad de disponer de tantos equipos conectados a diferentes puntos de Internet como se desee así como la imposibilidad de pagar licencias de software debe tenerse siempre en cuenta.

2.1.1.3 Modelo Genérico de la red

Para obtener resultados fiables en nuestro estudio se debe por un lado realizar un modelo de red genérico y parecido a la mayoría de los sistemas conectados a Internet. Por otro lado debemos realizar un diseño que nos

permita el control total de la red para su monitorización sin afectar al resto de los sistemas conectados.

2.1.1.4 Elementos de monitorización

Finalmente se debe realizar un análisis de los aspectos a monitorizar así como de las herramientas existentes puedan permitir su control y análisis.

2.2 Configuración

La elección del periodo de tiempo de análisis (una semana) viene determinada principalmente por tres factores:

2.2.1 Repetición de los patrones obtenidos.

La observación del tráfico en un día es insuficiente para poder extraer conclusiones plausibles, mientras que la observación en varios días puede permitir la correlación de resultados.

2.2.2 El tamaño de los datos en disco.

Cada semana se podrían llegar a generar hasta 18Gbytes de información, cantidad más que suficiente para analizar y realizar pruebas.

2.2.3 Seguridad.

La realización de este análisis requiere de una supervisión constante del sistema ya que cualquier ataque puede resultar exitoso y comprometer el sistema.

2.3 Herramientas de Monitoreo y Seguridad

En este apartado se realiza una breve explicación de las características principales y el rol asumido dentro de este análisis práctico de las diferentes herramientas de software elegidas.

Una vez decidido el objetivo del capítulo se procede a la selección de las herramientas necesarias para llevarlo a cabo. Los criterios basados para la selección de estos programas son los siguientes:

2.3.1 Plataforma de funcionamiento

Las herramientas elegidas deben funcionar perfectamente en la mayoría de plataformas existentes.

2.3.2 Licencia de uso

La licencia bajo la que se distribuya el software debe permitir su uso de forma libre y sin limitaciones contractuales en nuestro ámbito de estudio. Herramientas distribuidas bajo licencias GPL, GNU, BSD o similares serán las candidatas.

2.3.3 Continuidad del proyecto

Esto nos asegura que la herramienta seleccionada nos permitirá tener soporte de sus creadores en caso de necesidad.

2.3.4 Madurez del software

Se buscan herramientas estables que ya tengan desarrolladas varias versiones. Nuestro objetivo es el de evitar el uso de herramientas en fase de desarrollo o poco probadas que puedan introducir inestabilidad en el sistema.

2.4 Herramientas Utilizadas

2.4.1 TCPDUMP.

Es una herramienta que permite la auditoria y la adquisición del tráfico que circula por la red. La versión que se ha utilizado ha sido la 3.7.2.

Este software es gratuito y funciona sin problemas tanto en sistemas Unix como Windows. Pertenece a un proyecto estable que lleva varios años desarrollando software de calidad y se le considera como el referente principal en el análisis del tráfico de redes.

TCPdump permite el uso de filtros de tráfico (por protocolo, por dirección IP de origen o destino, por puerto de origen o destino...) así como diferentes opciones de captura tanto de las cabeceras de los paquetes como de los datos que transportan.

Otros programas de monitorización de redes capturan en mayor o menor medida el tráfico existente en la red pero únicamente a nivel de cabeceras

(como IPaudit) o a nivel de tamaño del paquete (como MRTG). TCPdump permite almacenar junto con la cabecera del datagrama los datos que transporta, lo que permite la reconstrucción total de las comunicaciones existentes en la red en análisis.

La instalación efectuada es la que se recomienda por defecto. Sin embargo se han de tener en cuenta los siguientes requisitos:

1. Necesita las librerías de interface LIBPCAP para su acceso a los dispositivos de red.
2. Para ejecutar todas las funcionalidades que proporciona esta herramienta se deben tener privilegios de administrador (*root*). Esto es debido principalmente a que accede directamente al dispositivo de red y lo configura en modo promiscuo para la captura de todo el tráfico.

2.4.1.1 Utilización Básica de TCPDUMP.

Para capturar el tráfico y enviarlo a la pantalla se utiliza la siguiente sintaxis

```
tcpdump -n -i < interface > -s < longitud >
```

Esta sintaxis hace uso de los siguientes indicadores

- -n indica a TCPdump que no resuelva las direcciones IP generando nombres de dominio, y los números de puerto generando nombres de servicio.
- -i < interface > indica a TCPdump que interfaz debe observar. Los dispositivos Ethernet de Linux suelen denominarse eth0
- -s < longitud > indica a TCPdump que parte del paquete debe registrar. Para Ethernet sin marcación de VLAN, 1515 bits suelen ser suficientes.

Por omisión, las versiones modernas de TCPdump ponen la interfaz que escucha en modo promiscuo, lo cual significa que se observará todo lo que pase en el puerto al que está conectado el dispositivo. El modificador `-p`, muy poco usado, desactiva este comportamiento.

Si no se le indica a TCPdump un valor de `-s < snaplen >`, este optará por capturar 68 bytes de forma predeterminada. Dado que el encabezado medio de IP tiene 20 bytes, y que el encabezado TCP tiene 20 bytes sin opciones, solo quedan 28 para datos de aplicación. Si están presentes 20 bytes o más de opciones de TCP, será casi imposible ver datos de aplicación, resultando frustrante el análisis de los datos capturados ya que no se ha capturado el contenido completo del paquete.

Por otra parte, capturar los datos de contenido completo es algo que resulta costoso. Si solo se necesitan los encabezados no es recomendable capturar los datos de aplicación.

2.4.1.2 Utilización de TCPDUMP para almacenar datos de contenido completo.

La sintaxis visualizada anteriormente se limitará a escribir en pantalla la interpretación que hace TCPdump de los paquetes. La sintaxis necesaria para almacenar paquetes en formato libpcap con tcpdump es sencilla según se muestra aquí:

```
tcpdump -n -i < interface > -s < snaplen > -w < capfile.lpc >
```

Si se añade el modificador `-w`, se envía el resultado TCPdump al fichero especificado. El sufijo `.lpc` indica que se trata de un fichero con formato libpcap.

Esta es la forma en la que se han capturado para cada interface los datos correspondientes al análisis y modelo de tráfico TCP/IP de la red de pruebas del DEEE, presentado mas adelante en este capítulo y en el capítulo 4.

```
tcpdump -n -i eth0 -s 1515 -w diaeth0
```

```
tcpdump -n -i eth1 -s 1515 -w diaeth1
```

```
tcpdump -n -i eth2 -s 1515 -w diaeth2
```

Cuando se ejecuta TCPdump, este seguirá escribiendo en la ubicación especificada hasta que se desborde la partición. Tiene sentido hacer uso de alguna estrategia de rotación o vigilancia de registro para evitar este final. Por lo tanto se debe orientar a TCPdump y programas similares hacia una partición dedicada. Si uno de los programas de monitorización se descontrola y llena la partición, no se verá afectado el resto del sistema.

2.4.1.3 Utilización TCPDUMP para leer datos de contenido almacenados.

Una vez que TCPdump ha capturado los paquetes, se puede leer los ficheros de seguimiento y ver lo que contienen. Se emplea para esto el modificador `-r`, junto con el nombre del fichero capturado, con el objeto de visualizar su contenido.

```
tcpdump -n -r diaeth0.lpc -c4
```

Se ha añadido el modificador `-c` para especificar que solo deben mostrarse cuatro paquetes. La alternativa es mandar los resultados a `more` o `less`, para mostrar solo una pantalla de resultados cada vez.

```
tcpdump -n -r diaeth0.lpc less
```

También se puede leer la información de acuerdo a cada protocolo es decir se puede filtrar la información según los requerimientos.

```
tcpdump -n -r diaeth0.lpc -c2 udp  
tcpdump -n -r diaeth0.lpc -c8 tcp
```

De esta manera se buscara únicamente los paquetes udp o tcp, y utilizando el modificador `-c` se indica la cantidad de paquetes de este tipo que se desea visualizar.

Los ficheros de log de salida generados por TCPdump son ficheros binarios que pueden ser visualizados/manipulados por otras herramientas como el Ethereal/Wireshark

2.4.2 TCPREPLAY.

TCPreplay es una herramienta que permite la reproducción del tráfico de red a partir de los logs generados por el programa TCPdump o compatibles. La versión utilizada en las pruebas realizadas es la 1.4.4.

La instalación efectuada es la que se recomienda por defecto. Sin embargo se han de tener en cuenta los siguientes requisitos:

1. Necesita las librerías de interface LIBPCAP para su acceso a los dispositivos de red.
2. Necesita las librerías de interface LIBNET para su acceso a los dispositivos de red.
3. Para ejecutar todas las funcionalidades que proporciona esta herramienta se deben tener privilegios de administrador (*root*).

2.4.2.1 Utilización TCPREPLAY para reproducir paquetes capturados con TCPDUMP.

TCPreplay suele utilizarse para enviar tráfico desde un sistema mientras una plataforma de monitorización intenta detectar ese tráfico. Para lograr reproducir y analizar el tráfico capturado mediante TCPdump se ha utilizado un host con sistema operativo LINUX (Centos 5.0) para reproducir el tráfico que circulo durante una semana en la red de pruebas del DEEE. El sistema encargado de la monitorización es un host con sistema operativo Windows XP en el cual se ha instalado el programa especializado en análisis y monitoreo de redes FLUKE NETWORK PROTOCOL EXPERT.

En este ejemplo de configuración el host que emplea TCPreplay envía el contenido del fichero diaeth0.lpc a un segmento de red en el cual esta escuchando el host que emplea FLUKE NETWORK PROTOCOL EXPERT. La computadora con sistema operativo LINUX se la ha configurado para que envíe su tráfico a través de la interfaz eth0. Obsérvese que si la interfaz que transmite el tráfico no posee una dirección IP, TCPreplay emitirá un mensaje de error. Por tal motivo se asignará una dirección IP arbitraria (192.168.1.2) para satisfacer a TCPreplay, de la misma manera se procede con el host con sistema operativo Windows XP al cual se le ha asignado la dirección IP (192.168.1.1). Para interrumpir el proceso de TCPreplay se pulsa Ctrl + C en el teclado.

Se debe tener cuidado para capturar el tráfico real cuando se utilice TCPreplay. Para obtener los resultados más limpios, se ha utilizado una LAN separada, sin tráfico de datos. Tres de las maneras de utilizar TCPreplay en una red basada en hardware implican conectar los dos sistemas mediante un concentrador, un conmutador o un cable cruzado. Para el análisis en este capítulo se han conectado los dos sistemas mediante un cable cruzado.

Se deben tener en cuenta también los tipos de tráfico que emitirán las estaciones de trabajo como resultado de un funcionamiento normal.

La interfaz eth0 de la computadora con LINUX produce el siguiente tráfico después de ponerse en situación inicial mediante la orden ifconfig.

```
16:03:35.627604 arp who-has 192.168.1.2 tell 192.168.1.1
```

Aparentemente, el controlador de eh0 actúa basándose en la intención de la NIC de localizar otros sistemas que utilicen ICMP con la versión 6 de IP. Este trafico ICMP no forma parte del fichero de captura diaeth0.lpc transmitido por TCPReplay, el tráfico ICMP esta causado de manera independiente por la NIC.

Además de las tres opciones basadas en hardware, se puede emplear un método exclusivamente de software, basado en un único sistema, para transmitir y observar paquetes mediante TCPReplay.

Además TCPReplay permite la posibilidad de reproducir estas secuencias tantas veces y a la velocidad que se desee, lo que permite simular distintos comportamientos.

TCPReplay además permite reproducir el tráfico capturado tantas veces y a la velocidad que se desee, lo que permite simular distintos comportamientos, así por ejemplo.

```
tcpreplay --topspeed --intf1=eth0 dayeth0.lpc
```

```
tcprepaly --mbps=10 --intf1=eth0 dayeth0.lpc
```

```
tcpreplay --pps=25 --intf1=eth0 dayeth0.lpc
```

- `--topspeed` es un modificador que indica que los paquetes a reproducirse serán transmitidos a la máxima velocidad, dependiendo del procesador del sistema y del medio por el cual se transmita
- `--mbps` es un modificador que hace referencia a los bit a transmitirse por segundo

- -- pps es un modificador que indica la cantidad de paquetes que se transmitirán por segundo

Para el análisis que compete a este capítulo se transmitirá el tráfico capturado mediante la siguiente instrucción.

Ejemplo de reproducción del tráfico capturado por TCPdump

```
tcpreplay - -intf1=eth0 dayeth0
```

```
tcpreplay - -intf1=eth0 dayeth1
```

```
tcpreplay - -intf1=eth0 dayeth2
```

Ya que lo que se desea es monitorear y analizar el tráfico en tiempo real, por esto no se debe modificar la velocidad ni la cantidad de paquetes a transmitirse.

2.4.3 Fluke Network Protocol Expert.

Para obtener medidas del comportamiento del tráfico, se utilizó el programa "FLUKE NETWORK PROTOCOL EXPERT".

Network Protocol Expert es una herramienta de monitoreo y seguridad exclusiva que realiza un seguimiento y diagnostica de forma activa los problemas en entornos TCP/IP, IPX y NetBIOS. Network Protocol Expert está diseñado para redes Ethernet LAN, identifica rápidamente si el problema se encuentra en un servidor, cliente, conmutador, enrutador o impresora gracias a la rápida detección y la clara visualización de la red.

2.4.3.1 Características.

Se consideran las siguientes.

2.4.3.1.1 Detección exhaustiva de dispositivos.

Network Inspector detecta rápidamente los dispositivos del dominio de difusión.

Con esta detección de dispositivos se obtiene la visibilidad de:

- Tipo/Nombres
- Direcciones: todas las direcciones IP asociadas al nodo y a la dirección MAC
- Servicios disponibles: conmutación, enrutamiento, correo electrónico, web e impresión
- Interfaces: velocidad y tipo
- Protocolos: IP, IPX y NetBIOS
- Configuración de la interfaz del dispositivo incluyendo velocidad, tipo, MTU, ranura y el conmutador más empleado y los puertos de enrutamiento, así como los dispositivos de cada puerto de conmutación.

La detección de dispositivos de Network Inspector también ofrece:

- Registro continuo de errores y cambios de los dispositivos de la red para aislar rápidamente los problemas
- Soluciones para los problemas de la red

3. ANÁLISIS DE RESULTADOS

Los resultados presentados a continuación hacen referencia al tráfico observado durante la semana del miércoles 30 de septiembre al viernes 9 de octubre de 2009. La presentación de los datos se desglosará en dos bloques.

Inicialmente se realiza una presentación de los datos mediante un informe diario que contendrá los datos más relevantes del día así como su análisis.

Dentro del informe diario se realizan dos presentaciones distintas de los datos obtenidos en nuestra red.

La primera clasificación dividirá en tráfico dirigido según los puertos a los que haga referencia. De esta forma se tiene los inferiores al 1024 denominados como conocidos (*well-know ports*) desglosados por servicios y la dirigida a puertos iguales o superiores al 1024.

La segunda clasificación agrupará los datagramas recibidos según el tipo de protocolo utilizado para su transmisión. De esta manera se puede establecer la cantidad de bytes por protocolo que circula en un determinado tiempo.

Además se puede observar las direcciones IP de origen y destino involucradas en la transmisión de información, dato adicional que determina cuál o hacia que IPs está relacionada la mayor cantidad de tráfico

Análogamente los protocolos en los que se subdivide el tráfico obtenido son los más usuales en Internet y corresponden a las capas de Red y de Transporte (TCP/IP).

- **TCP:** Protocolo orientado a conexión y fiable. Lo utilizan servicios como el SSH, TELNET, WWW...
- **UDP:** Protocolo no fiable. Lo utilizan servicios como el DNS, NFS...
- **ICMP:** Protocolo no fiable utilizado para la gestión y el control del flujo de las comunicaciones IP. Lo utilizan servicios como PING, Traceroute...
- **Otros:** Agrupa el resto de tráfico que no haga referencia a ninguno de los protocolos anteriores.

3.1 Análisis del primer día (miércoles 3 de octubre).

Como se menciona anteriormente al inicio de este capítulo, el objetivo principal del mismo es el de monitorear las 3 interfaces del servidor por las cuales circulan todo el tráfico de la red en análisis, así como utilizar las herramientas de monitoreo y seguridad adecuadas con la finalidad de extraer parámetros importantes del tráfico TCP/IP.

De esta manera lo que se pretende además de caracterizar el comportamiento normal del tráfico TCP/IP, es contar con las herramientas y la destreza que permitan detectar una posible anomalía de la red. Para llevar a cabo esta tarea se realiza un análisis detallado de cada una de las interfaces que conforman el servidor en un solo día, ya que para los demás días el análisis es el mismo.

3.1.1 Interfaz eth0

3.1.1.1 Iniciar el programa Fluke Network Protocol Expert.

Hay cuatro vistas principales de Network Protocol Expert, que incluyen lo siguiente:

- Vista resumida
- Vista detallada
- Vista de captura de Buffers de captura
- Vista de captura de archivos de captura

El programa se abre en **Summary View** (Vista resumida). Esta vista muestra varias ventanas que usa la herramienta. La ventana de **Resource Browser** (Navegador de recursos), en la esquina superior izquierda muestra el único dispositivo de monitoreo disponible, que es el Módulo NDIS 802.3 (NIC) del host.

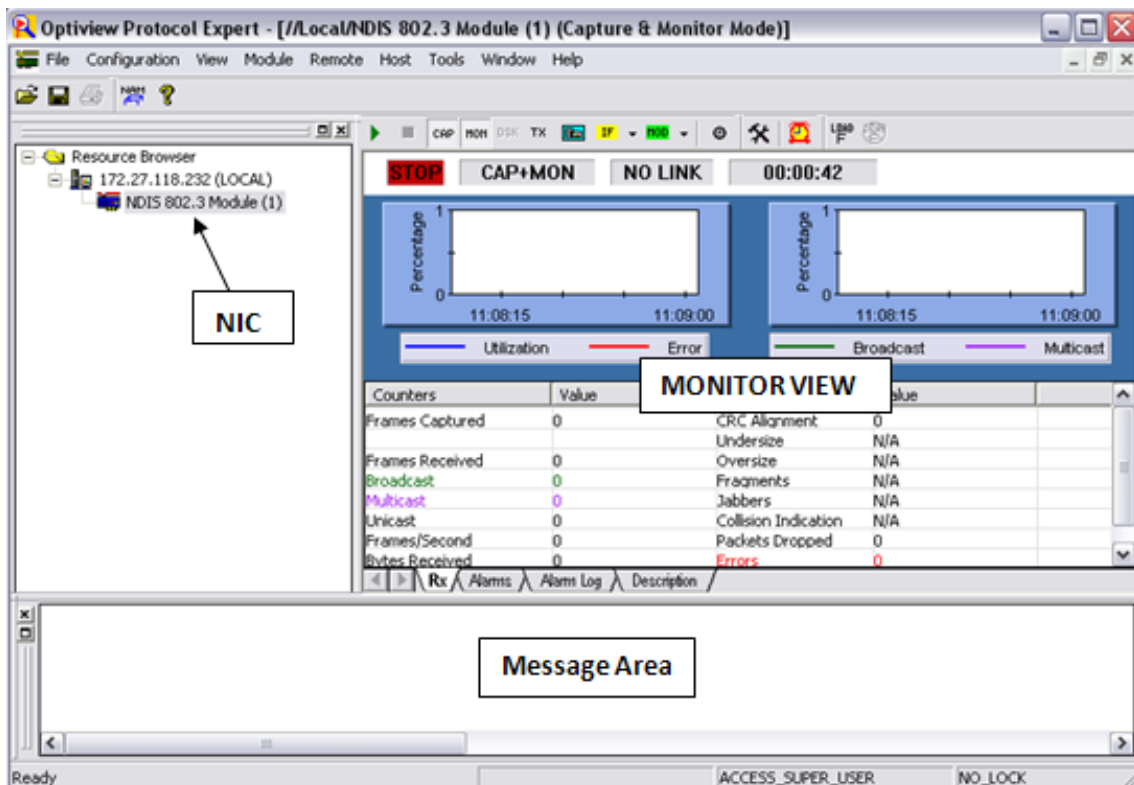



Figura. 3.2. Inicio del Programa Fluke Network Protocol Expert

Si hubiera Monitores de Medios de Protocolo, se mostrarían con los dispositivos de host asociados. El **Message Area** (Área de mensajes) en la parte inferior. **Stop** (Detener) en la esquina superior izquierda de la ventana de Vista de Monitoreo confirma que no se está realizando ningún monitoreo.

3.1.1.2 Inicio del proceso Monitoreo / Captura.

Para iniciar el proceso de monitoreo/captura, use el botón Inicio  o Module | Start (Módulo | Inicio) desde el menú principal de la ventana de monitoreo. El cuadro Utilization debe empezar a mostrar actividad, como en el gráfico siguiente:

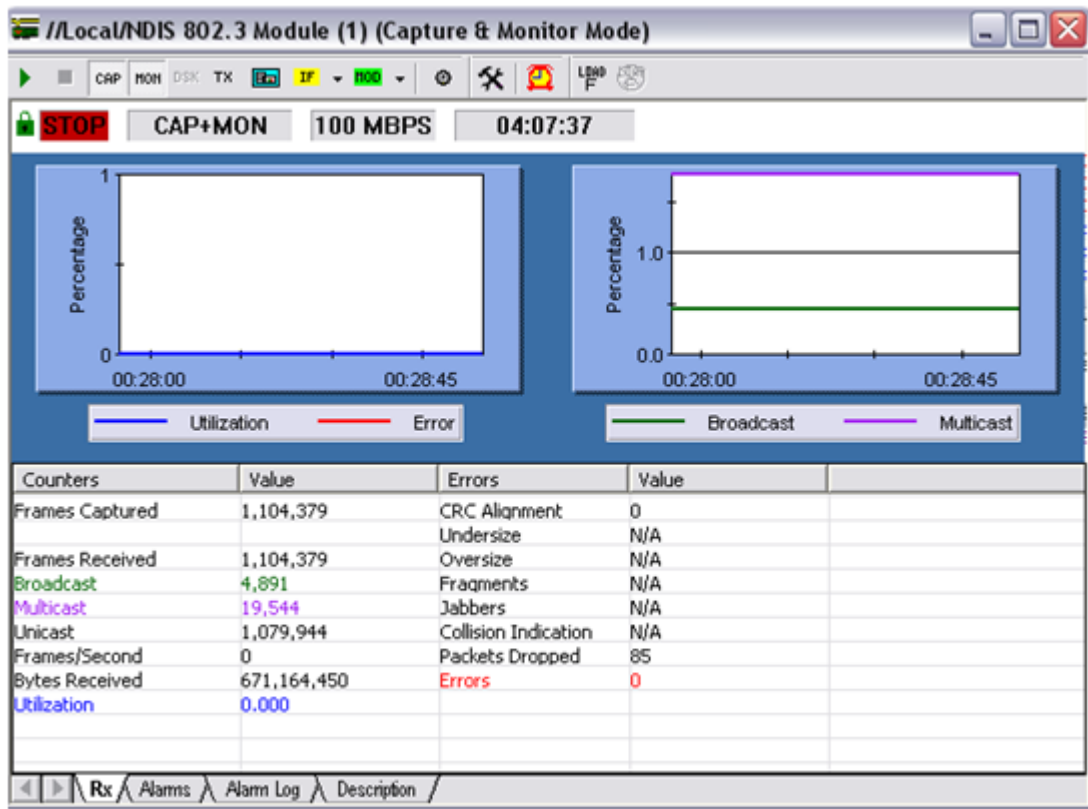



Figura. 3.3. Inicio del Proceso Monitoreo/Captura

La palabra **Arm** debe aparecer donde antes se veía **Stop**. Si se abre el menú **Module** (Módulo), observe que ahora **Stop**  es una opción, mientras que **Start** aparece difuminado.

Las fichas en la parte inferior de la ventana muestran los datos resultantes en una variedad de formas.

- **Transmit (Tx)** (Transmitir)
- **Alarms** (Alarmas)
- **Alarm Log** (Registro de alarmas).
- **Received (Rx)** (Recibidas), que indica las tramas capturadas y recibidas de **Broadcast** y **Multicast**.

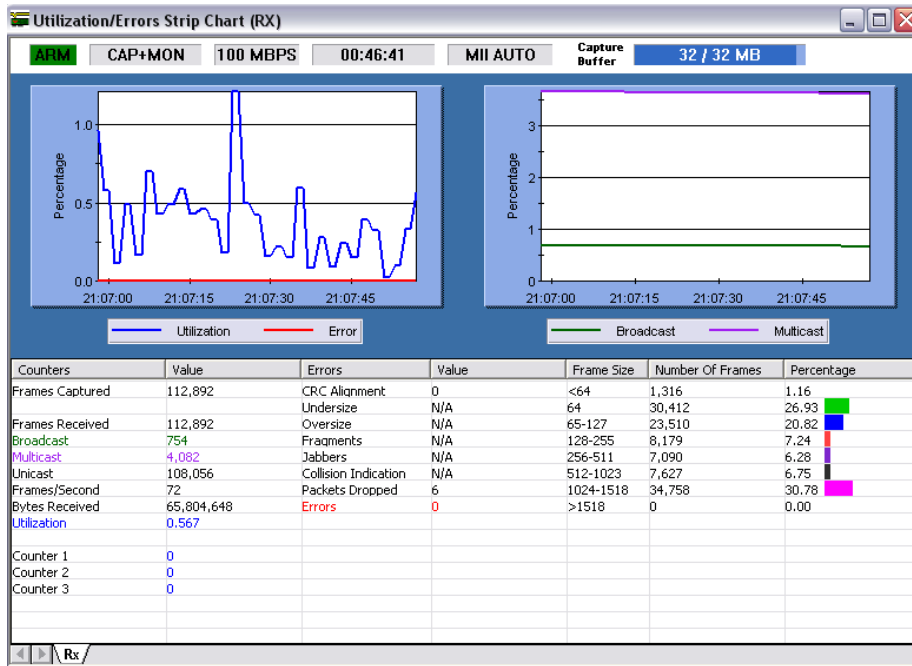


Figura. 3.4. Tramas capturadas y recibidas de Broadcast y Multicast

- **Description** (Descripción) muestra la dirección MAC, fabricante y modelo de la NIC. Muestra también cuáles son los Contadores de Errores activados.

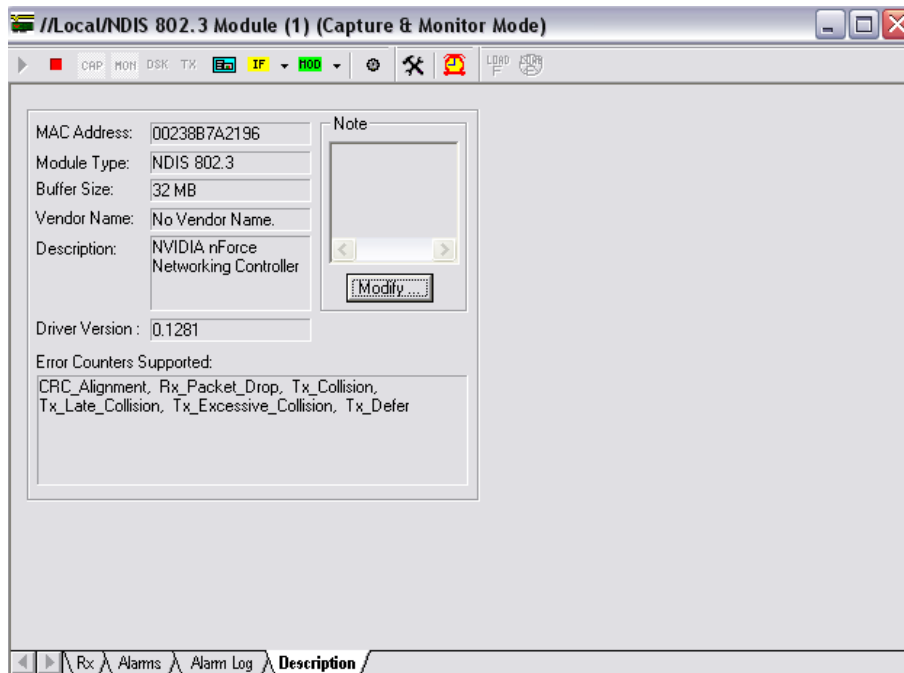



Figura. 3.5. Dirección MAC y Modelo de la NIC

Para ir a la ventana de **Detail View**  (Vista detallada) haga clic en el botón **Vista detallada** en la barra de herramientas o haga doble clic en cualquier parte en el diagrama Monitor View (Vista de monitoreo). Esto abre otra ventana que debe tener un aspecto similar a lo siguiente

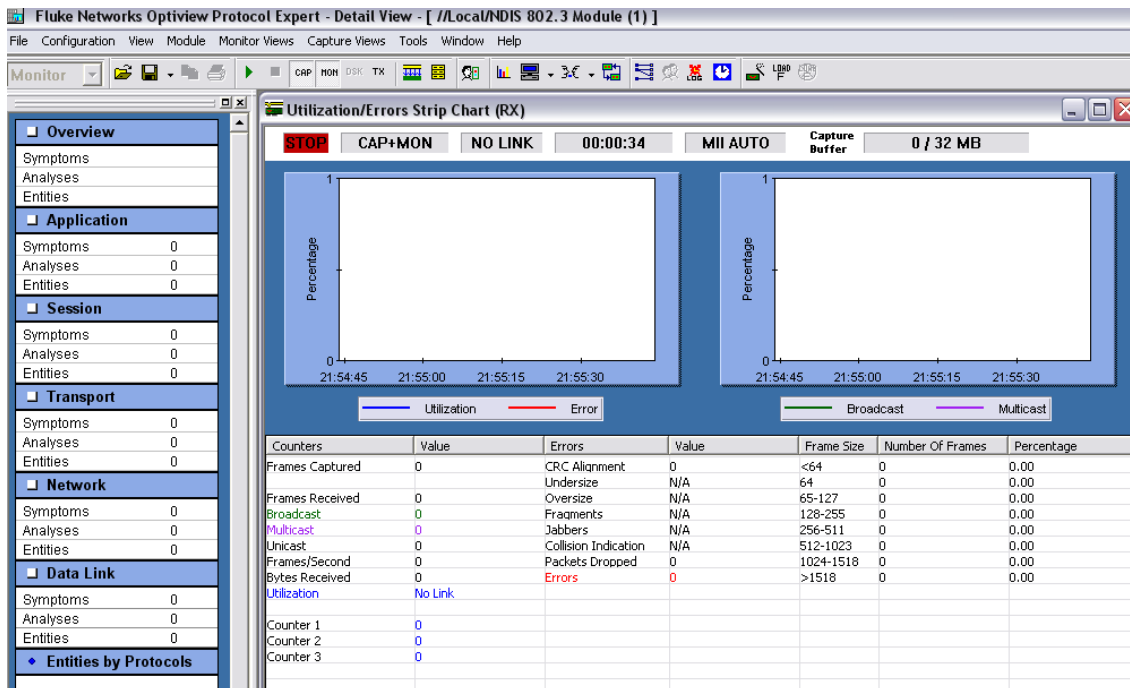



Figura. 3.6. Ventana Detail View

Nota: De ser necesario, se debe activar todas las barras de herramientas en el menú View (Ver).

Al principio, el resultado del diagrama es el mismo de antes. Sin embargo, hay muchas más opciones de barra de herramientas y menús que las que se ven en la Vista Resumida.

Al igual que todos los programas compatibles con Windows, al colocar el puntero del ratón sobre un botón, aparece una pantallita que identifica el propósito del botón. Al pasar el ratón sobre los botones, se verá que algunos aparecen difuminados. Esto significa que esta función no es apropiada para la situación actual.

3.1.1.3 Detener el proceso de captura.

Para detener la captura de tramas use el botón **Detener**  o Module | Stop (Módulo | Detener) del menú.

Una vez que se haya detenido la captura, haga clic en el botón **Ver captura**



La ventana resultante puede resultar algo abrumadora al principio. Maximice la ventana para ocultar cualquier otra ventana abierta en el fondo.

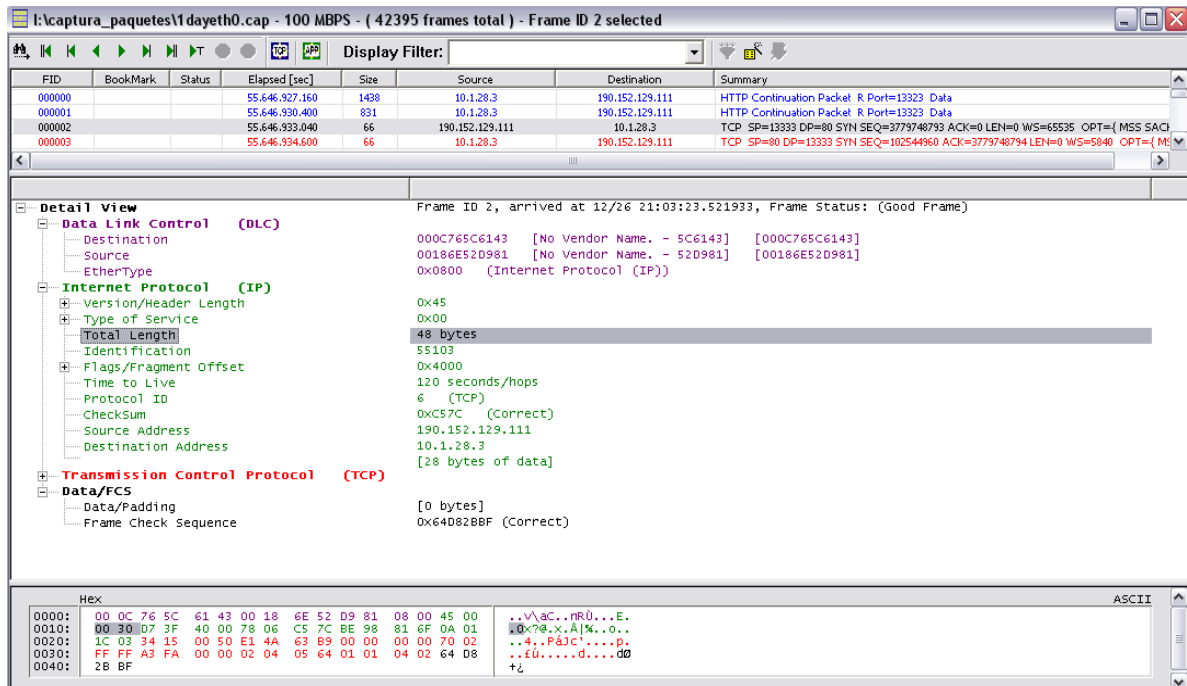


Figura. 3.7. Ventana de Captura

Se observa que hay tres ventanas horizontales abiertas. La ventana superior muestra los paquetes capturados. La ventana del medio muestra los detalles del paquete seleccionado en la ventana superior, y la ventana inferior muestra los valores HEX del paquete.

Al colocar el puntero del ratón sobre los bordes de las tres ventanas, aparece un desplazador de línea o flecha de dos cabezas. Esto permite cambiar la distribución de espacio para cada ventana.

Puede resultar conveniente agrandar la ventana del medio lo más posible, y dejar entre cinco y seis filas en cada una de las otras dos, como se muestra arriba en el Figura 2.6

Mire los paquetes enumerados en la ventana superior. Se deben encontrar paquetes TCP, UDP, ICMP, HTTP, DNS, ARP, y otros tipos de paquetes. Observe que cuando se seleccionan las filas en la ventana superior, cambia el contenido de las otras dos ventanas.

Seleccione la información en la ventana del medio, y observe que la vista HEX en la ventana inferior cambia para mostrar dónde se almacena esa información específica. En el ejemplo siguiente, seleccionar Source Address (IP) (Dirección origen) y se muestra los valores HEX del paquete.

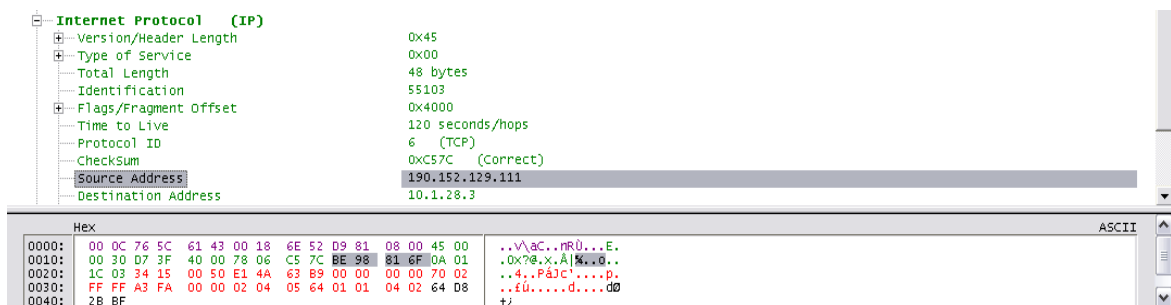


Figura. 3.8. Valores HEX del paquete

Observe también que el código de colores facilita la ubicación de la información de la ventana del medio en la ventana HEX.

En el ejemplo siguiente, con un paquete TCP, los datos en la sección Data Link Control (DLC) (Control de enlace de datos) son púrpura, mientras que la

sección de Internet Protocol (IP) (Protocolo Internet) es verde. Los valores correspondientes de HEX son de los mismos colores.

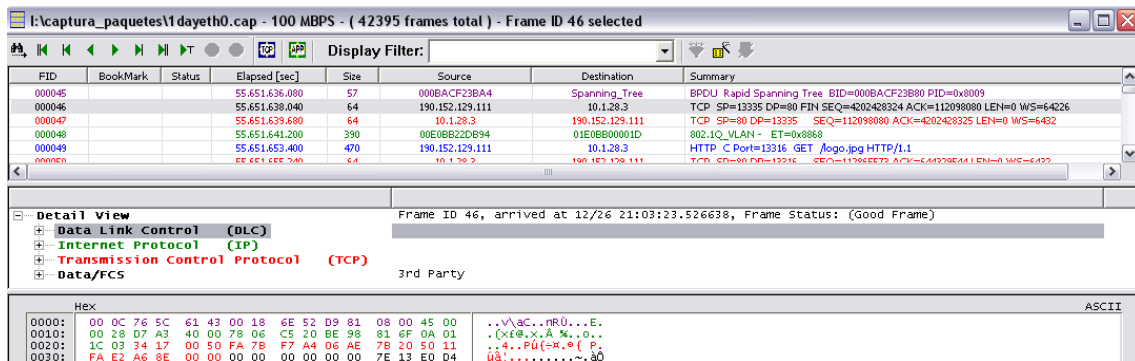


Figura. 3.9. Código de colores para la ubicación

Observe en el siguiente ejemplo que el **EtherType** es **0x0800**. Esto indica que es un paquete IP.

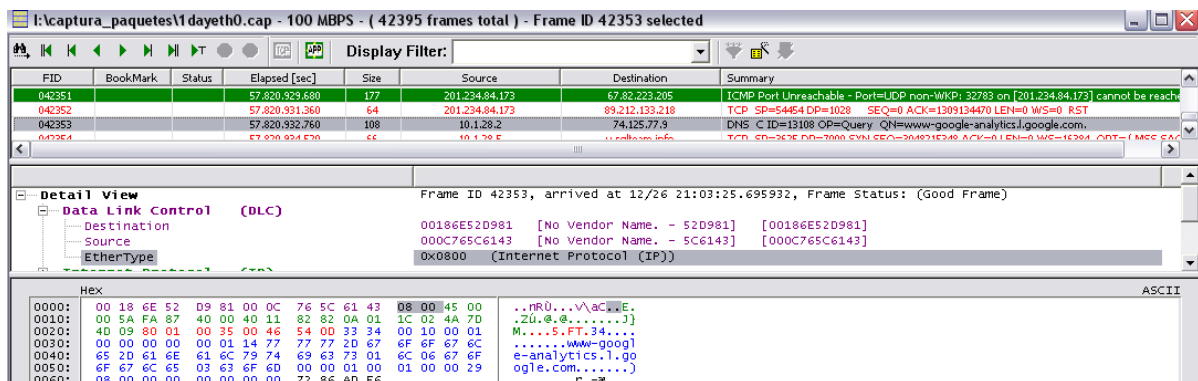


Figura. 3.10. EtherType (0x0800) Paquete IP

Observe las direcciones IP para los hosts Destination (Destino) y Source (Origen) así como los datos almacenados en la vista HEX.

FID	BookMark	Status	Elapsed [sec]	Size	Source	Destination	Summary
042350			57.820.928.000	144	dns1.counterpath.net	10.1.28.2	DNS R_ID=39284 QN=dns2.counterpath.net
042351			57.820.928.800	177	201.234.84.173	67.82.223.205	ICMP Port Unreachable - Port=UDP non-WiP: 32768 on [201.234.84.173] cannot be reached b...
042352			57.820.931.300	64	201.234.84.173	89.212.133.218	TCP SP=5464 DP=1028 SEQ=0 ACK=1309134470 LEN=0 WS=0 RST
042353			57.820.932.760	108	10.1.28.2	74.125.77.9	DNS C_ID=13108 OP=Query QN=www-google-analytics.l.google.com.
042354			57.820.934.520	66	10.1.28.5	u.sqteam.info	TCP SP=3625 DP=7000 SYN SEQ=3048215348 ACK=0 LEN=0 WS=16384 OPT={ MSS SACKpe...
042355			57.820.945.320	66	u.sqteam.info	10.1.28.5	TCP SP=7000 DP=3625 SEQ=0 ACK=3048215349 LEN=0 WS=16384 RST OPT={ MSS SACK...

Total Length: 90 bytes
 Identification: 64135
 Flags/Fragment Offset: 0x4000
 Time to Live: 64 seconds/hops
 Protocol ID: 17 (UDP)
 CheckSum: 0x8282 (Correct)
 Source Address: 10.1.28.2
 Destination Address: 74.125.77.9
 [70 bytes of data]

```

Hex
0000: 00 18 6E 52 09 81 00 0C 76 5C 61 43 08 00 45 00  ..nrÙ...v\ac...E.
0010: 00 5A FA 87 40 00 40 11 82 82 0A 01 1C 02 4A 7D  .Zü.@.@.....J]
0020: 40 09 80 01 00 35 00 46 54 00 33 34 00 10 00 01  M...S.FT.34....
0030: 00 00 00 00 00 01 14 77 77 77 2D 67 6F 6F 67 6C  .....WWW-googl
0040: 65 2D 61 6E 61 6C 79 74 69 63 73 01 6C 06 67 6F  e-analytics.l.go
0050: 6F 67 6C 65 03 63 6F 6D 00 00 01 00 01 00 00 29  ogle.com.....)
0060: 08 00 00 00 00 00 00 00 72 86 AD E6  .....r.-æ
  
```

Figura. 3.11. Direcciones IP y datos almacenados en la vista HEX

En el mismo ejemplo la siguiente sección en la ventana del medio es la información de **User Datagram Protocol (UDP)** (Protocolo del datagrama del usuario), que incluye los números de puerto UDP.

FID	BookMark	Status	Elapsed [sec]	Size	Source	Destination	Summary
042351			57.820.928.800	177	201.234.84.173	67.82.223.205	ICMP Port Unreachable - Port=UDP non-WiP: 32768 on [201.234.84.173] cannot be reached b...
042352			57.820.931.300	64	201.234.84.173	89.212.133.218	TCP SP=5464 DP=1028 SEQ=0 ACK=1309134470 LEN=0 WS=0 RST
042353			57.820.932.760	108	10.1.28.2	74.125.77.9	DNS C_ID=13108 OP=Query QN=www-google-analytics.l.google.com.
042354			57.820.934.520	66	10.1.28.5	u.sqteam.info	TCP SP=3625 DP=7000 SYN SEQ=3048215348 ACK=0 LEN=0 WS=16384 OPT={ MSS SACKpe...
042355			57.820.945.320	66	u.sqteam.info	10.1.28.5	TCP SP=7000 DP=3625 SEQ=0 ACK=3048215349 LEN=0 WS=16384 RST OPT={ MSS SACK...

Detail View: Frame ID 42353, arrived at 12/26 21:03:25.695932, Frame Status: (Good Frame)

- Data Link Control (DLC)
- Internet Protocol (IP)
- User Datagram Protocol (UDP)
 - Source Port: 32769
 - Destination Port: 53 (Domain Name Server)
 - Length: 70 bytes
 - Checksum: 0x5400 (Correct)
 - [62 bytes of data]

```

Hex
0000: 00 18 6E 52 09 81 00 0C 76 5C 61 43 08 00 45 00  ..nrÙ...v\ac...E.
0010: 00 5A FA 87 40 00 40 11 82 82 0A 01 1C 02 4A 7D  .Zü.@.@.....J]
0020: 40 09 80 01 00 35 00 46 54 00 33 34 00 10 00 01  M...S.FT.34....
0030: 00 00 00 00 00 01 14 77 77 77 2D 67 6F 6F 67 6C  .....WWW-googl
0040: 65 2D 61 6E 61 6C 79 74 69 63 73 01 6C 06 67 6F  e-analytics.l.go
0050: 6F 67 6C 65 03 63 6F 6D 00 00 01 00 01 00 00 29  ogle.com.....)
0060: 08 00 00 00 00 00 00 00 72 86 AD E6  .....r.-æ
  
```

Figura. 3.12. Información del User Datagram Protocol

La estructura de la ventana del medio cambia para cada tipo de paquete, al seleccionar diferentes tipos de paquete en la ventana superior, se puede observar los resultados en las otras dos ventanas. Se debe prestar atención especial al EtherType, número de puerto, así como las direcciones de origen y destino, que incluyen la capa de MAC y de Red.

3.1.1.4 Estimación de características y parámetros del tráfico capturado.

➤ Interfaz eth0

El miércoles 30 de septiembre del 2009 se registró un total de 42395 paquetes que circularon entre la mañana y tarde, dando un total de 26057,390 bytes que circularon por la interfaz eth0.

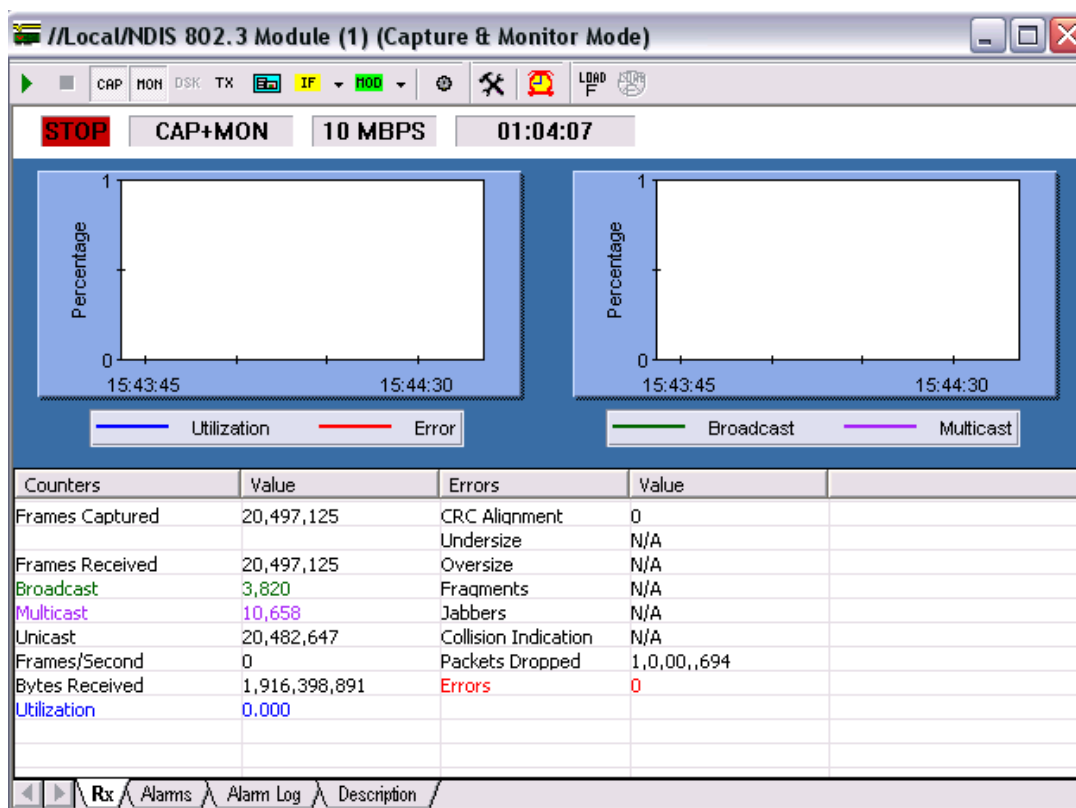


Figura. 3.13. Interfaz eth0 mañana

El desglose básico del tráfico clasificado por los principales servicios a los que hace referencia podemos observarlo en la figura 2.13

Haga clic en **CaptureViews** (Vista de Capturas), luego se escoge **Protocol Distribution** para ver una distribución de protocolos recibidos por la NIC. Al colocar el puntero del ratón sobre cualquier barra aparece un pequeño panel

de resumen en el que se muestra el protocolo en referencia y cantidad de paquetes y bits que corresponden al mismo, y el porcentaje que ocupa dentro de la distribución total de protocolos.

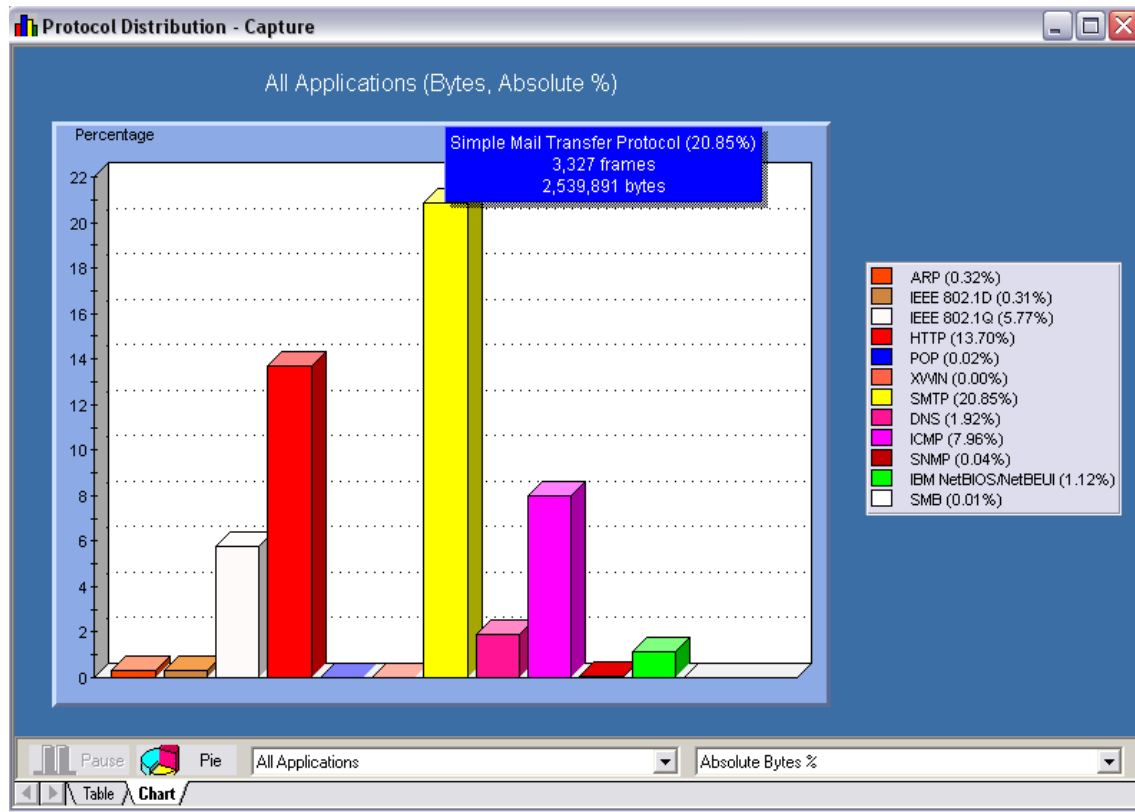
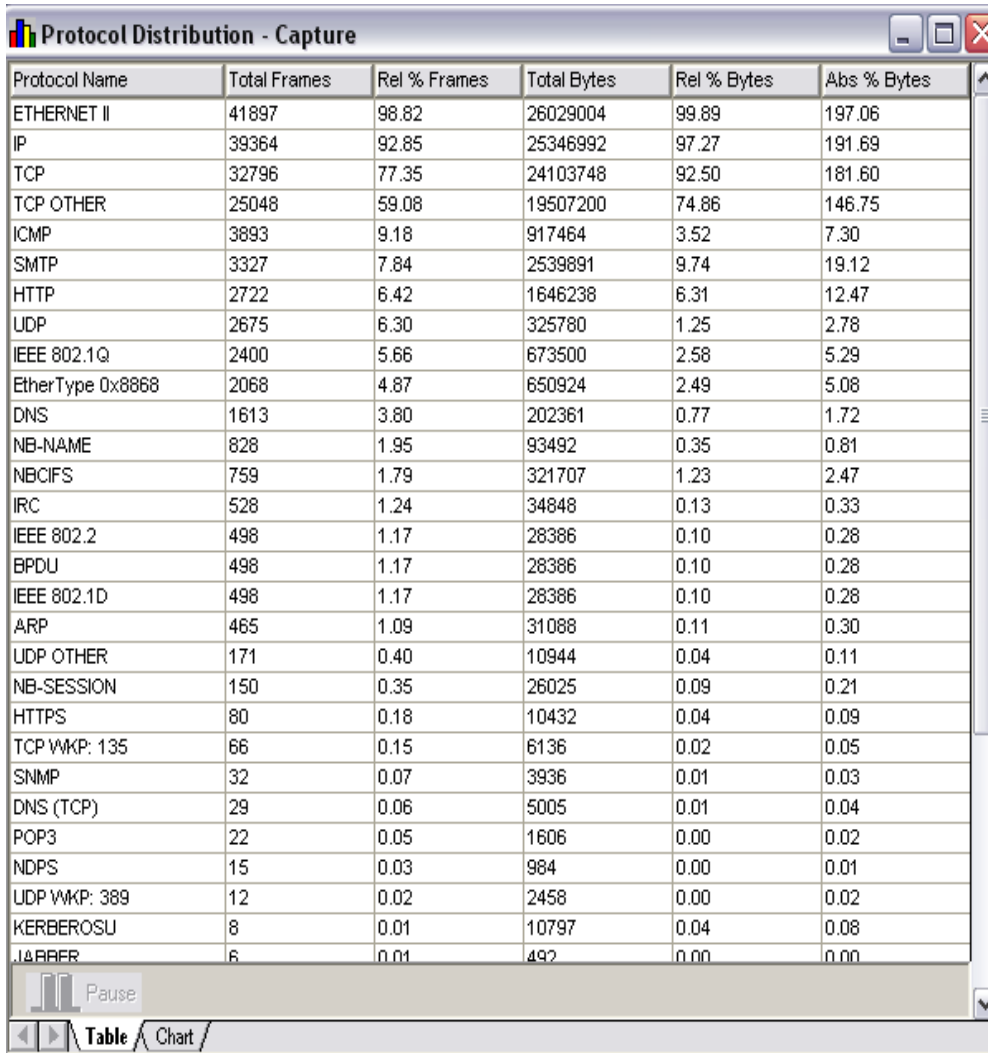


Figura. 3.14. Cantidad de bytes por Protocolo

Realice un clic en el botón **Table** que se encuentra en la parte inferior izquierda de la grafica y se obtiene la siguiente tabla que ilustra un resumen de la cantidad de Bytes por protocolo.



Protocol Name	Total Frames	Rel % Frames	Total Bytes	Rel % Bytes	Abs % Bytes
ETHERNET II	41897	98.82	26029004	99.89	197.06
IP	39364	92.85	25346992	97.27	191.69
TCP	32796	77.35	24103748	92.50	181.60
TCP OTHER	25048	59.08	19507200	74.86	146.75
ICMP	3893	9.18	917464	3.52	7.30
SMTP	3327	7.84	2539891	9.74	19.12
HTTP	2722	6.42	1646238	6.31	12.47
UDP	2675	6.30	325780	1.25	2.78
IEEE 802.1Q	2400	5.66	673500	2.58	5.29
EtherType 0x8868	2068	4.87	650924	2.49	5.08
DNS	1613	3.80	202361	0.77	1.72
NB-NAME	828	1.95	93492	0.35	0.81
NBCIFS	759	1.79	321707	1.23	2.47
IRC	528	1.24	34848	0.13	0.33
IEEE 802.2	498	1.17	28386	0.10	0.28
BPDU	498	1.17	28386	0.10	0.28
IEEE 802.1D	498	1.17	28386	0.10	0.28
ARP	465	1.09	31088	0.11	0.30
UDP OTHER	171	0.40	10944	0.04	0.11
NB-SESSION	150	0.35	26025	0.09	0.21
HTTPS	80	0.18	10432	0.04	0.09
TCP WKP: 135	66	0.15	6136	0.02	0.05
SNMP	32	0.07	3936	0.01	0.03
DNS (TCP)	29	0.06	5005	0.01	0.04
POP3	22	0.05	1606	0.00	0.02
NDPS	15	0.03	984	0.00	0.01
UDP WKP: 389	12	0.02	2458	0.00	0.02
KERBEROSU	8	0.01	10797	0.04	0.08
JABBER	6	0.01	492	0.00	0.00

Tabla. 3.1. Resumen de la cantidad de bytes por protocolo

3.1.1.5 Distribución de Paquetes de acuerdo al tamaño.

Para hacer uso de esta herramienta se da un clic en **Capture Views** (Vista de Captura) y se escoge **Frame Size Distribution** (Distribución de paquetes por Tamaño).

Con la ayuda de esta herramienta se visualiza la distribución de paquetes que circularon por la interfaz de acuerdo al tamaño, esta herramienta es de

gran utilidad ya que se puede tener una apreciación del tipo de tráfico, es decir se puede estimar si el tráfico que esta circulando por cierta interfaz se trata simplemente de actualizaciones entre los diferentes dispositivos que integran la red, o tráfico de datos que hace uso de una de las aplicaciones disponibles.

Frame Size (Bytes)	Number of Frames	Percentage
< 64	498	1.17
64	10,575	24.94
65 - 127	10,731	25.31
128 - 255	2,885	6.80
256 - 511	1,501	3.54
512 - 1023	1,302	3.07
1024 - 1518	14,903	35.15
> 1518	0	0.00
Total Bytes	26,057,390	
Total Frames	42,395	
Average Frame Size	615	
Average Throughput	2389028.2940 B/s	

Tabla. 3.2. Distribución de paquetes de acuerdo al tamaño

Para esta interfaz (eth0) se puede observar que el mayor porcentaje se encuentra en tráfico cuyo tamaño oscila entre (1024-151) Bytes, que de seguro con la interpretación que se tiene con el gráfico 2.13 (Cantidad de Bytes por Protocolo) una de las aplicaciones mas utilizadas y que mayor cantidad de datos posee es la de correo (SMTP-20.85%), seguido del protocolo (HTTP-13.7%).

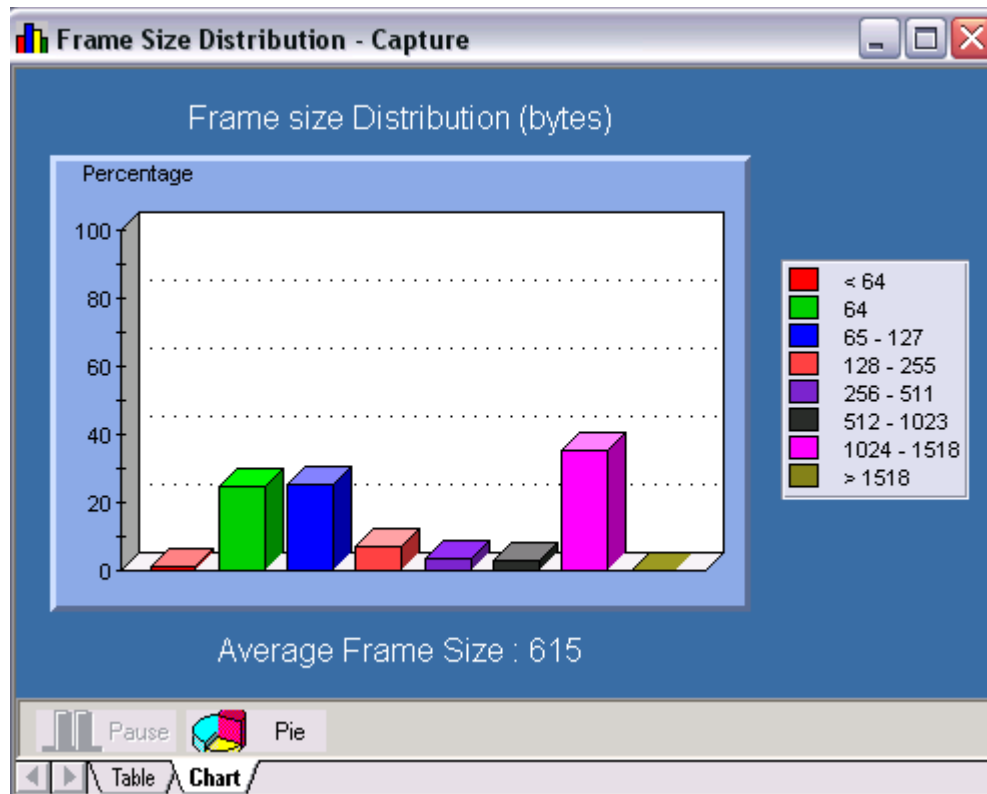


Figura. 3.15. Distribución de paquetes de acuerdo al tamaño

3.1.1.6 Captura de Paquetes según la dirección MAC.

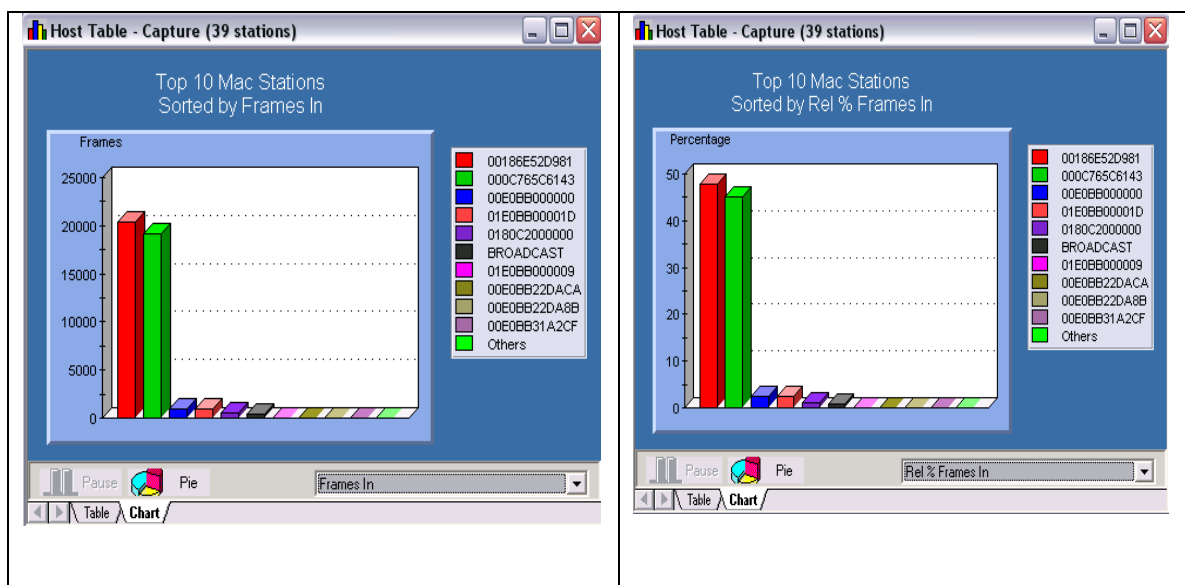
Para hacer uso de esta herramienta se da un clic en **Capture Views** (Vista de Captura) y se escoge **Host Table** (Tabla de Host de acuerdo a la dirección MAC).

Con la utilización de esta herramienta se puede obtener una lista de host de acuerdo a su dirección MAC. Esta lista de direcciones MAC permite al administrador de red determinar los host que mayor tráfico generan, creando una lista de confianza dentro de la red local, y así determinar de manera más prolija cuando un dispositivo de red existente o nuevo genera un porcentaje de tráfico que no corresponde a la operatividad normal de la red.

A continuación se muestra la cantidad de bytes de entrada y salida generados por cada host de acuerdo a su dirección MAC.

MAC Station Name	Frames In	Rel % Frames In	Frames Out	Bytes In	Bytes Out
00186E52D981	20325	47.94	19425	3631545	21741363
000C765C6143	19109	45.07	20361	21719927	3633849
00E0BB000000	1004	2.36	0	255016	0
01E0BB00001D	996	2.34	0	388440	0
0180C2000000	498	1.17	0	28386	0
BROADCAST	395	0.93	0	26608	0
01E0BB000009	50	0.11	0	3200	0
00E0BB29EC36	0	0.00	2	0	472
00238B7A2196	0	0.00	14	0	896
00E0BB31A2CF	1	0.00	0	238	0
00E0BB238CCB	1	0.00	0	238	0
00E0BB2A257E	0	0.00	1	0	236
00E0BB33CF37	1	0.00	0	238	0
00E0BB22DAAB	1	0.00	0	238	0
00E0BB1829E3	1	0.00	1	236	238
00E0BB22DB94	0	0.00	1046	0	391640
00E0BB2A2585	0	0.00	1	0	236
00E0BB238CA0	1	0.00	1	238	238
00E0BB234A55	0	0.00	1	0	238
00E0BB238C6E	0	0.00	1	0	238
00E0BB22D9FC	1	0.00	0	238	0
00E0BB272360	0	0.00	1	0	238
00E0BB237E3B	0	0.00	1006	0	255488
00E0BB22DACA	2	0.00	0	472	0
00E0BB234A56	0	0.00	1	0	238
00E0BB22DA8B	2	0.00	0	472	0
00E0BB22DA7A	1	0.00	0	238	0
00E0BB22DAAA	1	0.00	0	236	0
00E0BB23870D	0	0.00	1	0	238
00E0BB234B03	0	0.00	1	0	238
00E0BB2ADC9A	1	0.00	1	238	236
00E0BB22DA46	1	0.00	1	238	238
001EC1D3AA81	0	0.00	29	0	1972
00E0BB238C70	0	0.00	1	0	238
000BACF23BA4	0	0.00	498	0	28386
00E0BB2AD64A	0	0.00	1	0	236
00E0BB3547FF	1	0.00	0	236	0
00E0BB24DA41	1	0.00	0	236	0

Tabla. 3.3. Captura de paquetes según la dirección MAC



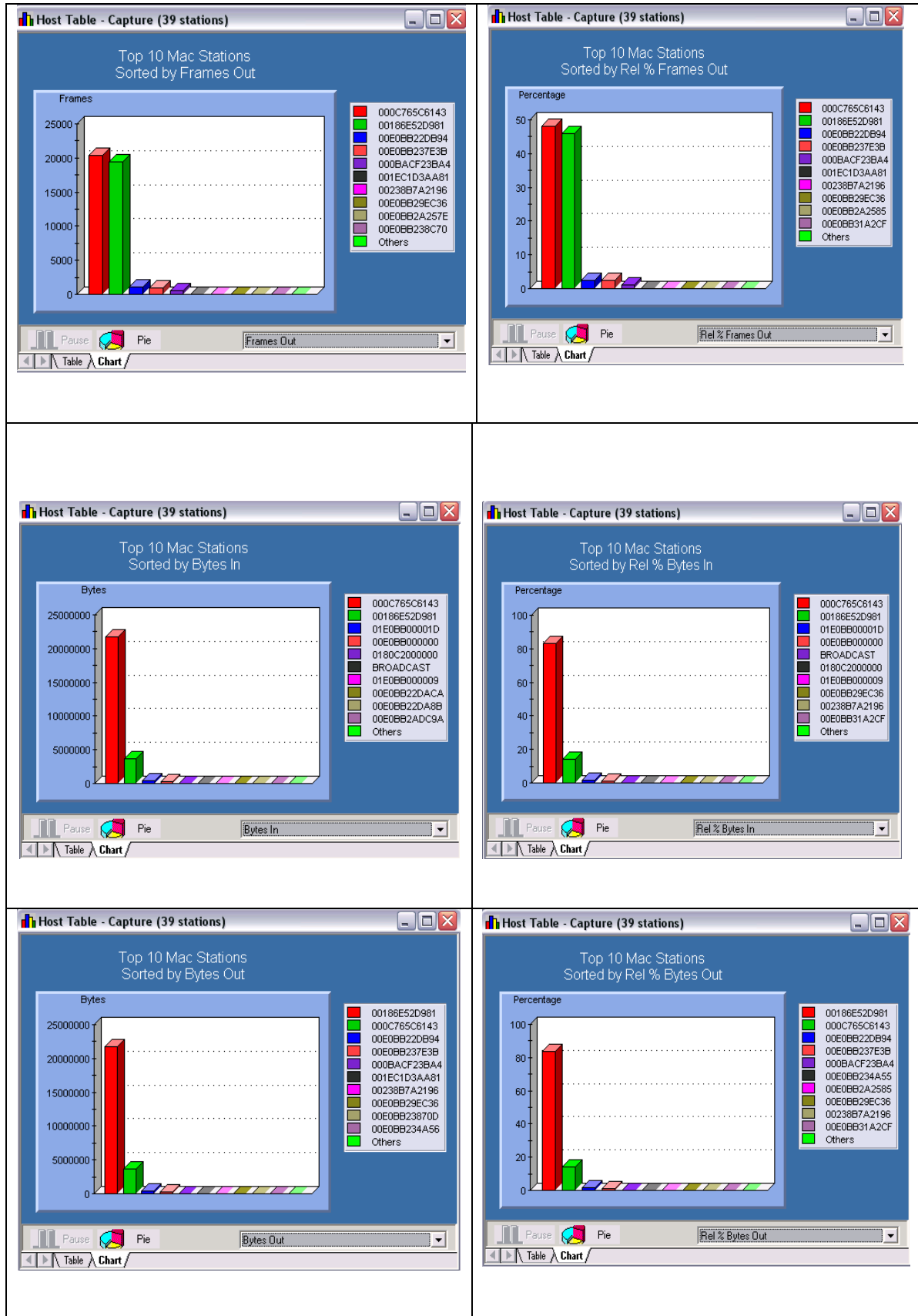


Figura. 3.16. Captura de paquetes según la dirección MAC

3.1.1.7 Captura de Paquetes según la dirección IP.

Para hacer uso de esta herramienta se da un clic en **Capture Views** (Vista de Captura) y se escoge **Network Layer Host Table** (Tabla de Host de acuerdo a la dirección IP).

Al igual que la tabla de host por dirección MAC, también puede resultar de utilidad clasificar e identificar las direcciones IP tanto públicas como privadas que mayor cantidad de tráfico genera dentro y fuera de la red Local, ya que de esta manera permite establecer una relación de confianza entre la dirección IP la cantidad y tipo de tráfico generado, teniendo de esta manera otra herramienta que permite validar si se trata o no de un posible ataque o intrusión.

A continuación se muestra una tabla que relaciona la dirección IP con los bytes de entrada y salida generados.

Network Station Name	Frames In	Rel % Frames In	Frames Out	Rel % Frames Out	Bytes In	Rel % Bytes In	Bytes Out	Rel % Bytes Out
10.1.0.112	6950	16.39	12949	30.54	521732	2.00	18609762	71.41
74.125.78.144	1559	3.67	1692	3.99	116640	0.44	2416641	9.27
10.1.28.3	2561	6.04	2941	6.93	2511253	9.63	1639142	6.29
10.1.28.5	16151	38.09	13381	31.56	19141521	73.45	1594522	6.11
201.234.84.173	0	0.00	3515	8.29	0	0.00	355344	1.36
10.1.0.134	306	0.72	332	0.78	33333	0.12	261153	1.00
10.1.0.101	1396	3.29	514	1.21	171833	0.65	82880	0.31
190.152.129.111	766	1.80	469	1.10	851409	3.26	52057	0.19
10.1.28.2	335	0.79	439	1.03	63095	0.24	39239	0.15
10.1.0.104	150	0.35	122	0.28	15099	0.05	19496	0.07
83.68.16.6	273	0.64	273	0.64	18018	0.06	18018	0.06
190.216.223.53	0	0.00	224	0.52	0	0.00	16576	0.06
69.42.216.219	159	0.37	159	0.37	10494	0.04	10494	0.04
208.110.87.83	142	0.33	142	0.33	9372	0.03	9372	0.03
190.245.213.146	218	0.51	68	0.16	294132	1.12	6094	0.02
69.42.216.215	114	0.26	114	0.26	7524	0.02	7524	0.02
85.55.34.83	37	0.08	35	0.08	3634	0.01	2664	0.01
192.188.58.50	18	0.04	15	0.03	8898	0.03	3045	0.01
10.1.0.106	60	0.14	58	0.13	4704	0.01	4534	0.01
201.37.147.73	35	0.08	34	0.08	4818	0.01	4828	0.01
10.1.53.211	1	0.00	0	0.00	82	0.00	0	0.00
192.188.81.212	1	0.00	0	0.00	82	0.00	0	0.00
93.3.166.225	3	0.00	3	0.00	246	0.00	246	0.00
189.234.30.173	1	0.00	0	0.00	101	0.00	0	0.00
188.126.64.3	10	0.02	10	0.02	640	0.00	640	0.00
188.126.64.2	10	0.02	10	0.02	640	0.00	640	0.00
190.174.201.54	3	0.00	3	0.00	246	0.00	246	0.00
62.128.52.191	11	0.02	0	0.00	704	0.00	0	0.00
187.147.22.65	16	0.03	0	0.00	1024	0.00	0	0.00
190.174.73.205	3	0.00	3	0.00	246	0.00	246	0.00
86.140.185.149	1	0.00	1	0.00	64	0.00	82	0.00
84.202.237.183	2	0.00	0	0.00	354	0.00	0	0.00
10.1.53.212	1	0.00	0	0.00	82	0.00	0	0.00
89.106.120.247	1	0.00	0	0.00	177	0.00	0	0.00
190.174.198.241	3	0.00	3	0.00	246	0.00	246	0.00
189.31.102.60	12	0.02	0	0.00	768	0.00	0	0.00
192.188.81.213	1	0.00	0	0.00	82	0.00	0	0.00
80.47.71.6	1	0.00	0	0.00	144	0.00	0	0.00

Tabla. 3.4. Captura de paquetes según la dirección IP

3.1.1.8 Numero de paquetes que circulan de acuerdo a la dirección IP.

En las siguientes gráficas se ilustra el porcentaje de bytes generados por cada una de las direcciones IP.

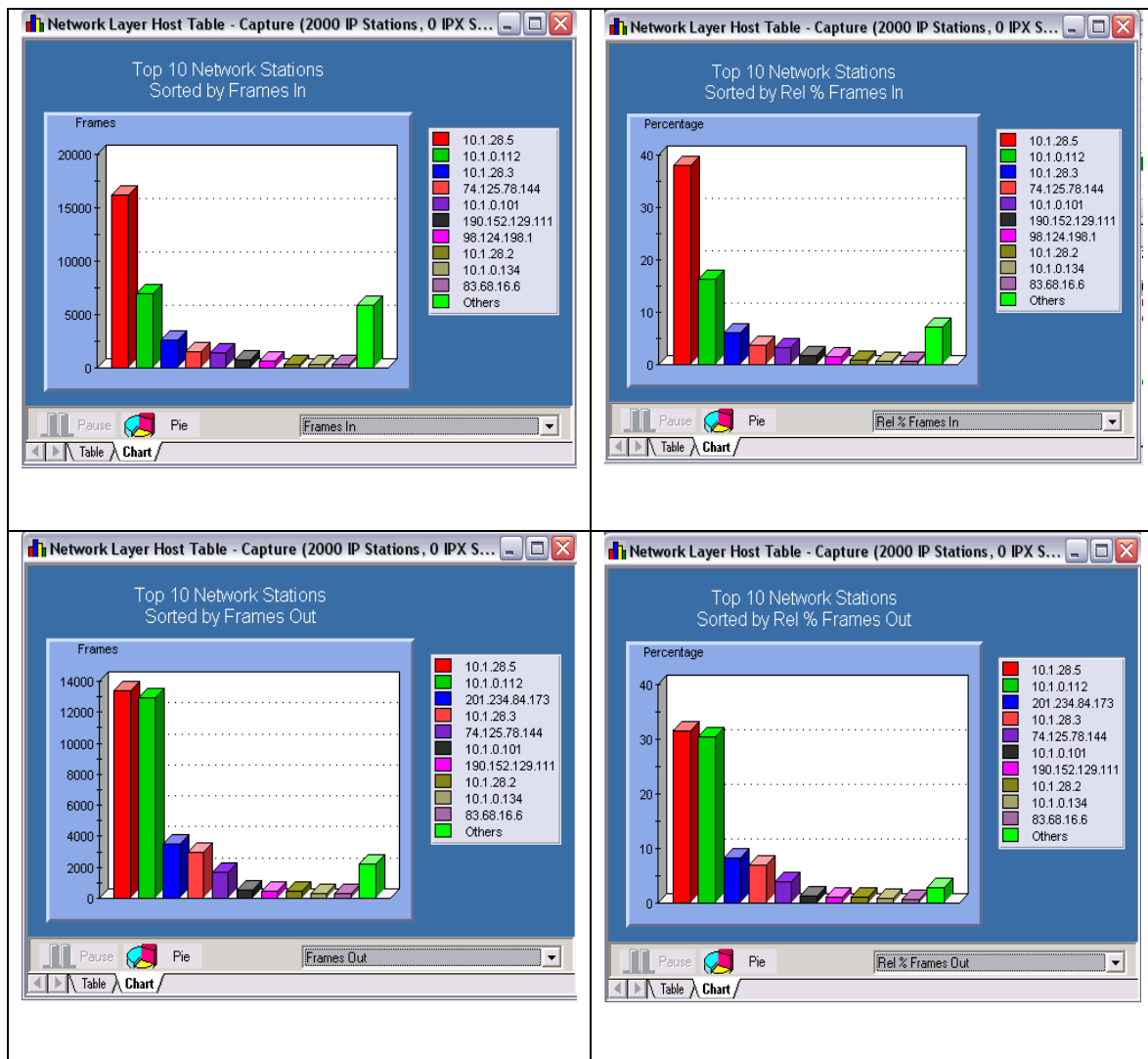


Figura. 3.17. Número de paquetes que circulan de acuerdo a la dirección IP

3.1.1.9 Numero de bytes que circulan de acuerdo a la dirección IP.

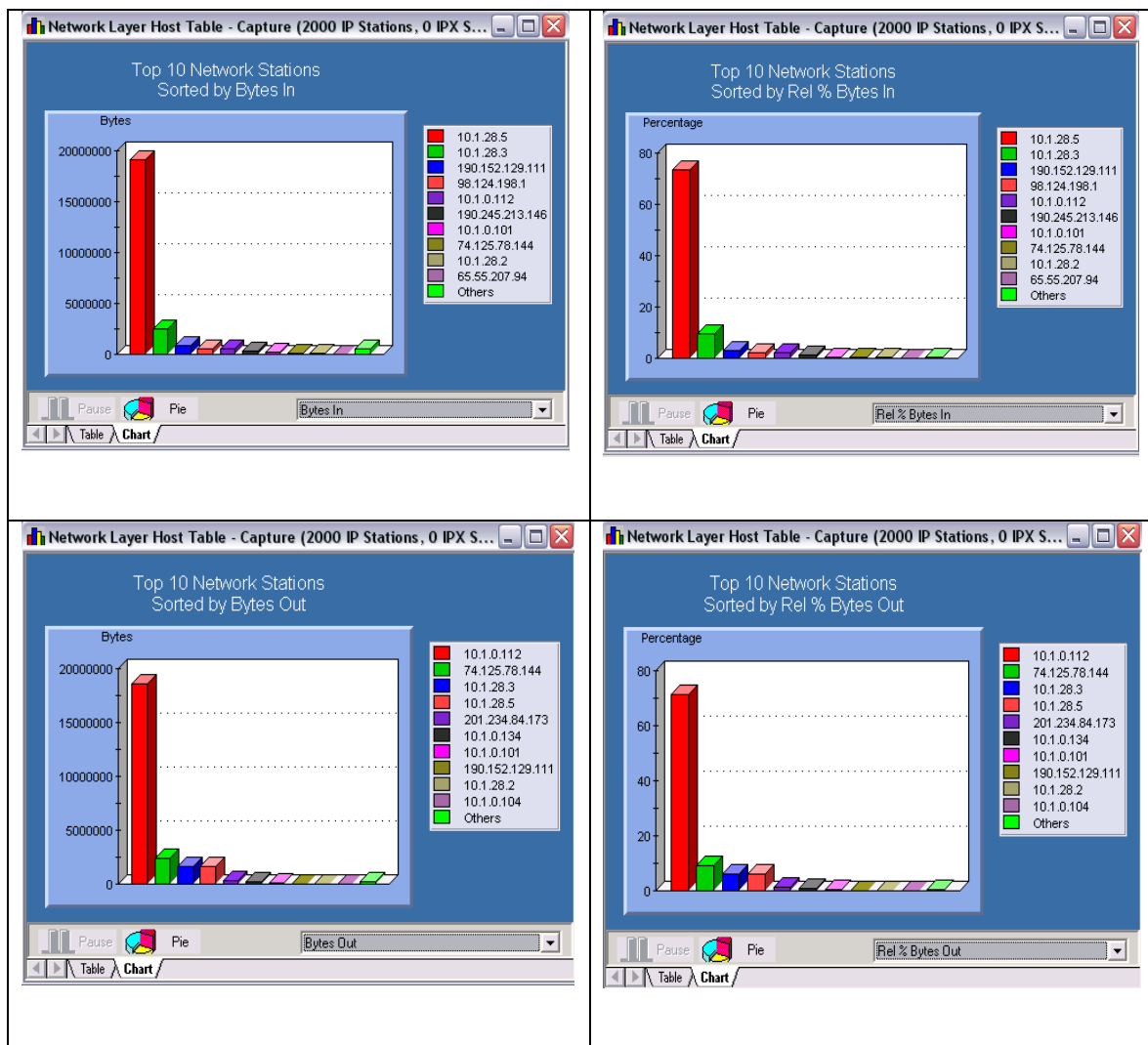


Figura. 3.18. Número de bytes que circulan de acuerdo a la dirección IP

3.1.1.10 Tiempo de respuesta.

Para hacer uso de esta herramienta se da un clic en **Capture Views** (Vista de Captura) y se escoge **Application Response Time** (Tiempo de Respuesta).

El tiempo de respuesta se define como el tiempo que pasa desde que se envía una comunicación y se recibe la respuesta. De esta manera con la

ayuda de esta herramienta se podrá tener la cantidad de bits que circulan por la red en un determinado tiempo, de acuerdo a cada protocolo. En el capítulo 4 se utilizarán estos datos en el modelo de tráfico.

La siguiente tabla muestra el tiempo promedio utilizado para cada conexión, se muestran tiempos mínimos y máximos con los cuales se puede establecer umbrales de tiempo dentro de los cuales se puede considerar a la comunicación confiable.

Server Name	Protocol	Minimum Time (ms)	Maximum Time (ms)	Average Time (ms)	Connections
81.7.134.249	DNS	0.444	0.444	0.444	2
195.59.44.134	DNS	0.426	0.426	0.426	2
192.112.36.4	DNS	0.425	0.425	0.425	2
77.72.229.252	DNS	0.413	0.413	0.413	2
66.218.167.1	DNS	0.394	0.394	0.394	2
91.200.16.100	DNS	0.393	0.393	0.393	2
208.44.108.138	DNS	0.372	0.372	0.372	2
17.254.0.59	DNS	0.361	0.361	0.361	2
17.112.144.59	DNS	0.350	0.350	0.350	2
192.43.172.30	DNS	0.335	0.335	0.335	2
62.41.78.201	DNS	0.331	0.331	0.331	2
192.12.94.30	DNS	0.038	2.993	1.137	6
4.23.59.51	DNS	0.031	0.031	0.031	2
216.239.36.10	DNS	0.027	0.861	0.444	4
209.59.13.227	DNS	0.025	0.025	0.025	2
216.239.38.10	DNS	0.024	0.851	0.568	6
17.112.144.50	DNS	0.021	0.021	0.021	2
204.2.178.133	DNS	0.021	0.858	0.417	6
204.2.249.6	DNS	0.020	3.447	1.165	6
65.203.229.15	DNS	0.020	0.020	0.020	2
128.242.118.131	DNS	0.020	1.312	0.588	6
66.135.215.5	DNS	0.019	0.019	0.019	2
192.42.93.30	DNS	0.019	0.019	0.019	2
66.135.207.138	DNS	0.019	0.019	0.019	2
66.218.167.3	DNS	0.018	0.018	0.018	2
66.218.167.2	DNS	0.018	0.018	0.018	2
192.31.80.30	DNS	0.018	0.415	0.151	6
203.105.65.195	DNS	0.017	0.017	0.017	2
192.26.92.30	DNS	0.017	3.405	0.963	8
64.215.156.30	DNS	0.017	0.017	0.017	2
204.2.249.38	DNS	0.016	0.020	0.018	4
204.2.249.5	DNS	0.016	0.016	0.016	2

Tabla. 3.5. Tiempo promedio utilizado para cada conexión

3.1.1.11 Resumen de toda la actividad generada por la interfaz eth0

➤ Capa de Transporte

Transport	
Symptoms	1349
Analyses	7
Entities	409

Expert Symptom	Expert Summary
TCP Fast Retransmission	In 1 ms (< 100 ms) between [10.1.28.5]/[TCP non-WKP: 4346] and [70.71.227.29]/[TCP non-WKP: 55741]
TCP Fast Retransmission	In 0 ms (< 100 ms) between [10.1.28.5]/[TCP non-WKP: 4347] and [79.109.19.41]/[TCP non-WKP: 52007]
TCP Window Exceeded	Data length of 1146 bytes exceeds last window size of 256
TCP Window Exceeded	Data length of 298 bytes exceeds last window size of 256
TCP Window Exceeded	Data length of 550 bytes exceeds last window size of 255
TCP Window Exceeded	Data length of 860 bytes exceeds last window size of 253
TCP Window Exceeded	Data length of 1460 bytes exceeds last window size of 256
TCP Window Exceeded	Data length of 1460 bytes exceeds last window size of 256
TCP Window Exceeded	Data length of 298 bytes exceeds last window size of 256
TCP Window Exceeded	Data length of 550 bytes exceeds last window size of 255

Symptoms	
TCP Fast Retransmission	1

Host 1: 10.1.28.5	
Packets In	16151
Packets Out	13381
Octets In	19141521
Octets Out	1594522
Non Unicast Packets Out	0
Associated MAC Address	000C76:5C6143

Host 2: 70.71.227.29	
Packets In	2
Packets Out	2
Octets In	164
Octets Out	164
Non Unicast Packets Out	0
Associated MAC Address	00186E:52D981

Tabla. 3.6. Resumen de toda la actividad generada por la interfaz eth0

➤ Symptoms:

Dentro de los síntomas que se pueden observar en el flujo normal de datos se menciona el mecanismo de control de congestión de tráfico TCP Fast Retransmission que consiste en lo siguiente:

➤ **Mecanismos Fast Retransmit y Fast Recovery**

Los mecanismos de retransmisión y recuperación rápida, intentan optimizar y ajustar los algoritmos de control de congestión de tráfico, para adecuar el protocolo TCP a situaciones de congestión y pérdidas en general, y ayudarlo así a recuperarse rápidamente de las mismas.

Los mecanismos de congestión actúan a posteriori, es decir, una vez detectada la pérdida o presunta pérdida de un segmento. Esto representa que no se intentará solucionar el problema hasta transcurridos RTO segundos.

➤ **Retransmisión**

La primera de las mejoras propuesta que está presente en la mayoría de implementaciones, aprovecha la recepción de **reconocimientos duplicados** (aquellos que consecutivamente reconocen al mismo segmento, es decir contienen el mismo número de secuencia). De esta forma se activa la retransmisión de un segmento presuntamente perdido antes de que expire su temporizador de retransmisión.

Este algoritmo se conoce como **Retransmisión Rápida** ó **Fast Retransmit**.

La recepción de un ACK duplicado puede ser debida las siguientes situaciones:

- La red desordena paquetes y, en consecuencia, es posible que el receptor haya enviado un reconocimiento duplicado ante la llegada de un segmento que no sigue la secuencia normal.

- Hay que tener en cuenta que todas las implementaciones modernas de TCP generan, inmediatamente, un ACK duplicado al recibir un segmento fuera de orden.
- Se ha perdido algún segmento de datos tanto por errores del canal o por problemas de congestión. En este caso el TCP recibe segmentos fuera de orden y en consecuencia genera ACK duplicados.
- Se ha producido un pico de retardo en la red, es decir, el tiempo de ida y vuelta de un paquete se ha incrementado repentinamente provocando la expiración del temporizador de retransmisión del emisor con la consiguiente retransmisión del segmento conflictivo. Puesto que el segmento en cuestión ya había sido recibido correctamente, la recepción de su réplica provoca la generación de un ACK duplicado. Esta situación se producirá básicamente en casos de congestión.

El emisor, sin embargo, no sabe a cuál de estas razones responde la recepción del ACK duplicado. Experimentalmente se ha comprobado que no resulta apropiado precipitar la retransmisión ante la recepción del primer ni el segundo reconocimiento duplicado en redes que desordenan paquetes, ya que puede tratarse de un simple problema de reordenación de rápida solución. Por esta razón, la mayoría de implementaciones de TCP activan el mecanismo de Retransmisión Rápida al recibir el **tercer ACK duplicado**. Cuando éste llega, el emisor no espera a la expiración del temporizador sino que retransmite, inmediatamente, el segmento que demanda el reconocimiento e inicia el algoritmo de Inicio Lento.

Este mecanismo, pensado para situaciones de congestión, es también muy adecuado para las situaciones en las que la pérdida es debida a errores en la comunicación, ya que en el mejor de los casos se retransmitirá el paquete perdido antes que en implementaciones sin este mecanismo. **Es por tanto adecuado tanto para redes fijas como móviles.**

Además de la retransmisión de los datos perdidos, los algoritmos contra la congestión reducen la ventana de transmisión, para dar tiempo a la red que se recupere de ésta.

El algoritmo combinado de Inicio Lento y Prevención de la Congestión da una solución muy drástica a los problemas de congestión, pudiendo degradar de forma innecesaria el comportamiento de TCP frente a estas situaciones. El hecho de cerrar la ventana de congestión a un único segmento después de la pérdida de datos y empezar un inicio lento no parece adecuado ni en los casos de congestión ni en los casos de errores.

Analizando con más detalle el mecanismo propuesto anteriormente, puede ampliarse basándonos en el hecho que un ACK duplicado no sólo indica que ha habido un problema en la red, sino que también confirma que un paquete ha abandonado la red y ha sido recibido correctamente por el TCP en el otro extremo.

Esto es así puesto que el receptor sólo puede generar un ACK como respuesta a la llegada de un paquete. No hay necesidad, por tanto, de poner en marcha Inicio Lento y reducir así drásticamente el número de paquetes inyectados en la red.

Este algoritmo se conoce como *Retransmisión Rápida con Recuperación Rápida* ó *Fast*

Retransmit with Fast Recovery, y opera como sigue:

- Cuando llega el tercer ACK duplicado se retransmite el segmento perdido y:
- $ssthresh = cwnd / 2$
- $cwnd = ssthresh + 3$
- Con cada nuevo ACK duplicado recibido, se incrementa $cwnd$ en una unidad y se transmite un nuevo paquete si lo permite el valor de $cwnd$.

- Cuando se recibe el primer ACK que reconoce nuevos datos, se asigna a *cwnd* el valor de *ssthresh* (es decir la mitad del valor que tenía la ventana de congestión cuando se produjo la congestión).

➤ Entities

Muestra las conexiones TCP generadas.

Client	Server	Protocol	First Frame	Last Frame	Pkts In	Pkts Out	Octs In	Octs Out
190.152.129.111:13323	10.1.28.3:80	TCP	0	24	3	4	2213	80
190.152.129.111:13332	10.1.28.3:80	TCP	12	31	4	4	553	551
190.152.129.111:13335	10.1.28.3:80	TCP	33	47	5	5	1417	583
10.1.28.5:4346	70.71.227.29:55741	TCP	37	67	2	2	88	88
10.1.28.5:4347	79.109.19.41:52007	TCP	25	60	2	2	88	88
10.1.28.5:4348	201.233.31.163:57844	TCP	51	52	1	1	44	44
10.1.28.5:4349	190.179.153.96:56880	TCP	69	70	1	1	44	44
10.1.0.112:8080	10.1.28.5:54574	TCP	575	579	2	3	40	977
10.1.0.112:8080	10.1.28.5:54575	TCP	581	589	5	4	197	1238
10.1.0.112:8080	10.1.28.5:54576	TCP	590	602	7	6	1474	1840
10.1.0.112:8080	10.1.28.5:54577	TCP	603	626	12	12	2423	10284

The screenshot shows a network analysis tool interface. On the left, a tree view displays the hierarchy: TCP: 13323 <-> 80, 190.152.129.111, 00186E:52D981, 10.1.28.3, and 000C76:5C6143. On the right, there are three panels:

- 190.152.129.111:13323 <-> 10.1.28.3:80**:

Packets In	3
Packets Out	4
Octets In	2213
Octets Out	80
Start Time	Sat Dec 26 21:03:22
Stop Time	Sat Dec 26 21:03:22
Duration (s)	0
- 190.152.129.111:13323**:

Max Window Size	65535
Min Window Size	65535
Retransmission	0
Zero Window Size	0
Analyses	0
Symptoms	0
Acknowledgments	2
Max Ack Time (ms)	0.046
Min Ack Time (ms)	0.043
Avg Ack Time (ms)	0.045
- 10.1.28.3:80**:

Max Window Size	6432
Min Window Size	6432
Retransmission	0
Zero Window Size	0
Analyses	0
Symptoms	0
Acknowledgments	1
Max Ack Time (ms)	0.002
Min Ack Time (ms)	0.002
Avg Ack Time (ms)	0.002

Tabla. 3.7. Resumen de toda la actividad generada por la interfaz eht0

➤ Capa de Red

Network	
Symptoms	1550
Analyses	0
Entities	2000

Symptoms

Expert Symptom	Expert Summary
ICMP Port Unreachable	Port=UDP non-WKP: 32783 on [201.234.84.173] cannot be reached by [82.232.48.218], SA=[201.234.84.173]
ICMP Port Unreachable	Port=UDP non-WKP: 22850 on [201.234.84.173] cannot be reached by [124.149.112.234], SA=[201.234.84.173]
ICMP Port Unreachable	Port=UDP non-WKP: 21928 on [201.234.84.173] cannot be reached by [189.234.30.173], SA=[201.234.84.173]
ICMP Port Unreachable	Port=UDP non-WKP: 32783 on [201.234.84.173] cannot be reached by [84.202.237.183], SA=[201.234.84.173]
ICMP Port Unreachable	Port=UDP non-WKP: 54454 on [201.234.84.173] cannot be reached by [89.106.120.247], SA=[201.234.84.173]
ICMP Port Unreachable	Port=UDP non-WKP: 54454 on [201.234.84.173] cannot be reached by [80.47.71.6], SA=[201.234.84.173] DA=
ICMP Port Unreachable	Port=UDP non-WKP: 54454 on [201.234.84.173] cannot be reached by [86.76.151.173], SA=[201.234.84.173]
ICMP Port Unreachable	Port=UDP non-WKP: 54454 on [201.234.84.173] cannot be reached by [70.45.161.136], SA=[201.234.84.173]
ICMP Port Unreachable	Port=UDP non-WKP: 32060 on [201.234.84.173] cannot be reached by [81.191.63.60], SA=[201.234.84.173] D
ICMP Port Unreachable	Port=UDP non-WKP: 32783 on [201.234.84.173] cannot be reached by [217.120.159.45], SA=[201.234.84.173]

Symptoms	
ICMP Port Unreachable	1

Host 1: 201.234.84.173	
Packets In	0
Packets Out	3515
Octets In	0
Octets Out	355344
Non Unicast Packets Out	0
Associated MAC Address	000C76:5C6143

Host 2: 82.232.48.218	
Packets In	1
Packets Out	0
Octets In	172
Octets Out	0
Non Unicast Packets Out	0
Associated MAC Address	00186E:52D981

Tabla. 3.8. Resumen de toda la actividad generada por la interfaz eth0

Dentro de los síntomas que presenta la red de pruebas se puede observar el mensaje de error de ICMP Port Unreachable este mensaje consiste en lo siguiente:

➤ ICMP Destination Unreachable

ICMP Destination Unreachable es un tipo de paquete ICMP cuya función es transportar un mensaje que es generado por un enrutador, y se envía al host de origen, que recibe el mensaje emitido por el enrutador.

El mensaje en sí significa que este router considera inalcanzable el destino al que quiere llegar el host.

Si se recibe de parte del host de destino, significa que el protocolo que se intentó acceder no está activo en aquel momento.

El campo Type tiene el valor 3. El campo código contendrá el valor de 3 de acuerdo a los siguientes valores:

- 0 network unreachable
- 1 Host unreachable
- 2 Protocol unreachable
- 3 port unreachable
- 4 Fragmentation needed, but Do not fragment bit set
- 5 Source route failed
- 6 Destination network unknown
- 7 Destination host unknown
- 8 Source host isolated error (military use only)
- 9 The destination network is administratively prohibited
- 10 The destination host is administratively prohibited
- 11 The network is unreachable for Type Of Service
- 12 The host is unreachable for Type Of Service
- 13 Communication administratively prohibited (administrative filtering prevents packet from being forwarded)
- 14 Host precedence violation (indicates the requested precedence is not permitted for the combination of host or network and port)
- 15 Precedence cutoff in effect (precedence of datagram is below the level set by the network administrators)

➤ **Entities**

Muestra las conexiones IP generadas.

Network Station	Last MAC Station	Protocol	First Frame	Last Frame	Pkts In	Pkts Out	Octs In	Octs Out
10.1.28.3	000C76:5C6143	IP	0	39690	2561	2941	2511253	1639142
190.152.129.111	00186E:52D981	IP	0	19467	766	469	851409	52057
10.1.28.5	000C76:5C6143	IP	5	42382	16151	13381	19141521	1594522
10.1.0.101	00186E:52D981	IP	5	42325	1396	514	171833	82880
201.234.84.173	000C76:5C6143	IP	6	42383	0	3515	0	355344
88.246.64.147	00186E:52D981	IP	6	566	3	0	192	0
82.232.48.218	00186E:52D981	IP	9	9	1	0	172	0
188.24.67.239	00186E:52D981	IP	17	27910	12	0	768	0
190.162.113.187	00186E:52D981	IP	20	34519	8	0	512	0
79.109.19.41	00186E:52D981	IP	25	120	3	3	246	246
80.101.60.192	00186E:52D981	IP	32	18849	9	0	576	0

Host: 10.1.28.3	
Packets In	2561
Packets Out	2941
Octets In	2511253
Octets Out	1639142
Non Unicast Packets Out	0
Associated MAC Address	000C76:5C6143

Protocol						
Protocol	Frm <-	Frm ->	Bytes <-	Bytes ->	F. Frm	L. Frm
UDP	4	0	256	0	21682	21686
UDP OTHER	4	0	256	0	21682	21686
TCP	2557	2938	2510997	1638872	0	39690
SMTP	1727	1600	2419305	120586	3369	39690
HTTP	819	1327	90870	1517582	0	38624
TCP WKP: 135	3	3	246	192	188	18696
NB-SESSION	2	2	164	128	15760	15810
NBCIFS	6	6	412	384	1697	16269
ICMP	0	3	0	270	21684	21688

Tabla. 3.9. Resumen de toda la actividad generada por la interfaz eth0

De la misma manera que se realizó el análisis con la interfaz eth0, utilizando las herramientas del software FLUKE NETWORK PROTOCOL EXPERT se procederá con las otras 2 interfaces (eth1, eth2) extrayendo los conceptos mas importantes que permitan posteriormente definir el comportamiento del tráfico que circula por cada interfaz.

3.1.2 Interfaz eth1

3.1.2.1 Cantidad de Bytes por Protocolo.

Protocol Name	Total Frames	Rel % Frames	Total Bytes	Rel % Bytes	Abs % Bytes
IEEE 802.1D	52	0.15	2964	0.01	0.00
eDonkey	24	0.07	10677	0.04	0.00
ARP	220	0.65	14080	0.05	0.00
BPDU	52	0.15	2964	0.01	0.00
CPFI	1	0.00	149	0.00	0.00
ETHERNET II	33331	99.83	25582245	99.98	5.13
UDP	283	0.84	53482	0.20	0.01
MSNMS	38	0.11	4465	0.01	0.00
IP	33111	99.17	25568165	99.93	5.13
UDP WVKP: 80	1	0.00	149	0.00	0.00
CDP	2	0.00	718	0.00	0.00
X.25	18	0.05	1730	0.00	0.00
SSL	31	0.09	4391	0.01	0.00
ICMP	508	1.52	35357	0.13	0.01
UDP OTHER	274	0.82	52301	0.20	0.01
TCP	32320	96.80	25479326	99.58	5.11
TDS	17	0.05	1777	0.00	0.00
NFS	74	0.22	54453	0.21	0.01
TACACS+	18	0.05	1736	0.00	0.00
TCP WVKP: 68	13	0.03	1343	0.00	0.00
HTTP	458	1.37	268890	1.05	0.05
TCP WVKP: 85	809	2.42	703478	2.74	0.14
POP3	19	0.05	1286	0.00	0.00
SUNRPC	74	0.22	54453	0.21	0.01
JABBER	4	0.01	354	0.00	0.00
IEEE 802.2	52	0.15	2964	0.01	0.00
TCP OTHER	29982	89.80	23907001	93.43	4.79
Teredo	6	0.01	786	0.00	0.00
IEEE SNAP	2	0.00	718	0.00	0.00
HTTPS	27	0.08	5830	0.02	0.00

Tabla. 3.10. Cantidad de bytes por protocolo

3.1.2.2 Cantidad de Bytes

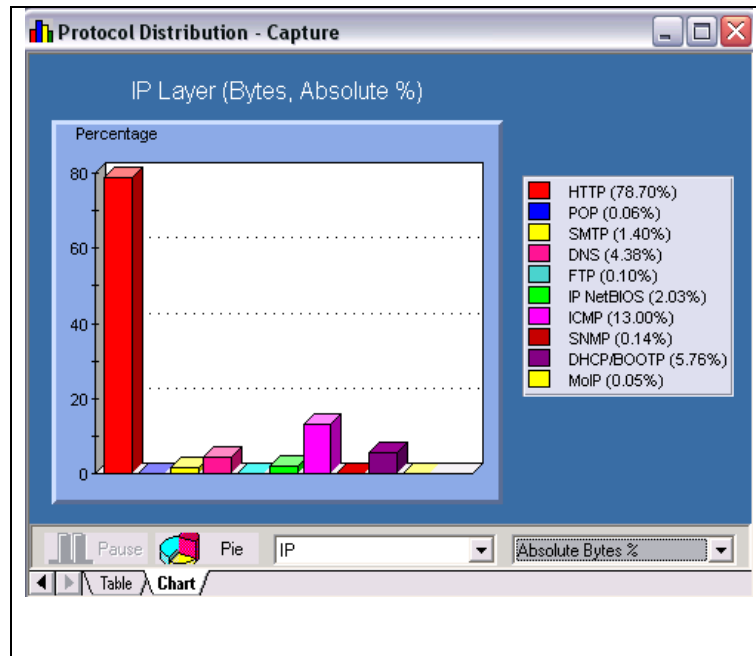


Figura. 3.19. Cantidad de bytes por protocolo

3.1.2.3 Distribución de Paquetes de acuerdo al tamaño

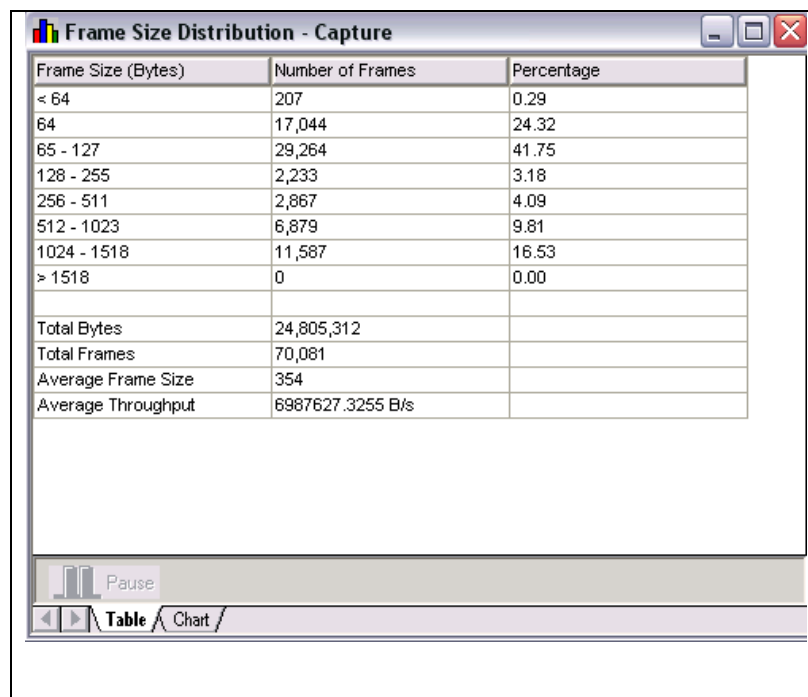


Tabla. 3.11. Distribución de paquetes de acuerdo al tamaño

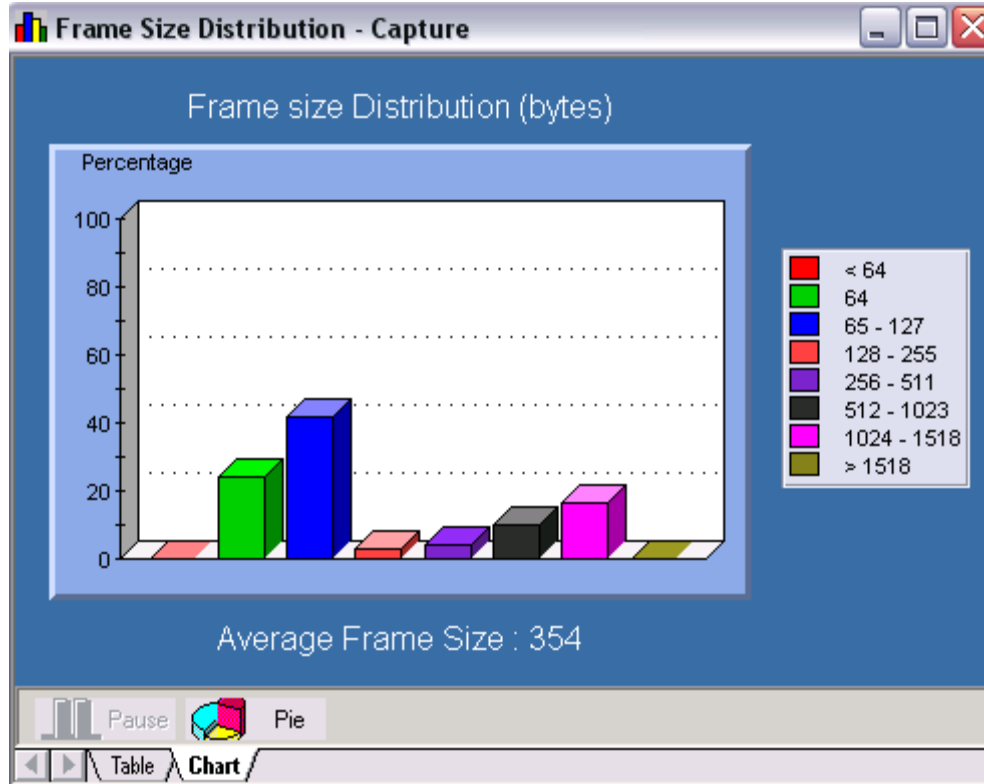


Figura. 3.20. Distribución de paquetes de acuerdo al tamaño

3.1.2.4 Captura de Paquetes según la dirección MAC

MAC Station Name	Frames In	Rel % Frames In	Frames Out	Bytes In	Bytes Out
0013F7486D6F	34350	49.01	32453	11254954	12914702
00238B40B943	8456	12.06	12091	986740	7616716
000C765C6144	5483	7.82	8169	1639602	2005747
001CC016DA.A8	5427	7.74	4224	7749729	300931
001CC017E649	4281	6.10	1170	397123	178994
000C765C619D	3217	4.59	387	241729	84086
001F1601195B	1950	2.78	1993	214815	241549
BROADCAST	1748	2.49	0	258865	0
002185102E83	1361	1.94	2418	1081771	295867
001F16514B0C	758	1.08	1290	264344	215180
001E68897E60	444	0.63	928	75774	160213
01E0BB00001D	414	0.59	0	161460	0
00E0BB000000	418	0.59	0	106172	0
001B24580A21	226	0.32	599	59079	108402
Spanning_Tree	207	0.29	0	11799	0
00E0BB31.A6F5	187	0.26	0	11968	0
001CC0064D7B	157	0.22	1051	78240	80132
00507FF05BC3	158	0.22	471	54812	114254
001CC017E625	151	0.21	306	25157	30186
01005E7FFFFA	125	0.17	0	44643	0
001CC017F3E6	109	0.15	136	17868	21624
090009000067	82	0.11	0	9894	0
0013202912EE	77	0.10	89	5847	6878
001E6826E62B	47	0.06	153	6509	40609
33330000000C	36	0.05	0	20094	0
0019E3436706	35	0.04	74	5821	10180
0019D16B640B	26	0.03	43	1842	5313
002185103465	23	0.03	45	2177	4940
001CC017E2F0	27	0.03	30	2609	2811
01005E40988F	28	0.03	0	4732	0
01E0BB000009	21	0.02	0	1344	0
001CC0177075	16	0.02	28	1207	3724
333300010003	16	0.02	0	1488	0
01005E0000FC	16	0.02	0	1168	0
333300010002	10	0.01	0	1576	0
0180C200000E	14	0.01	0	2240	0
00E0BB237E3B	0	0.00	421	0	106880

Tabla. 3.12. Captura de paquetes según la dirección MAC

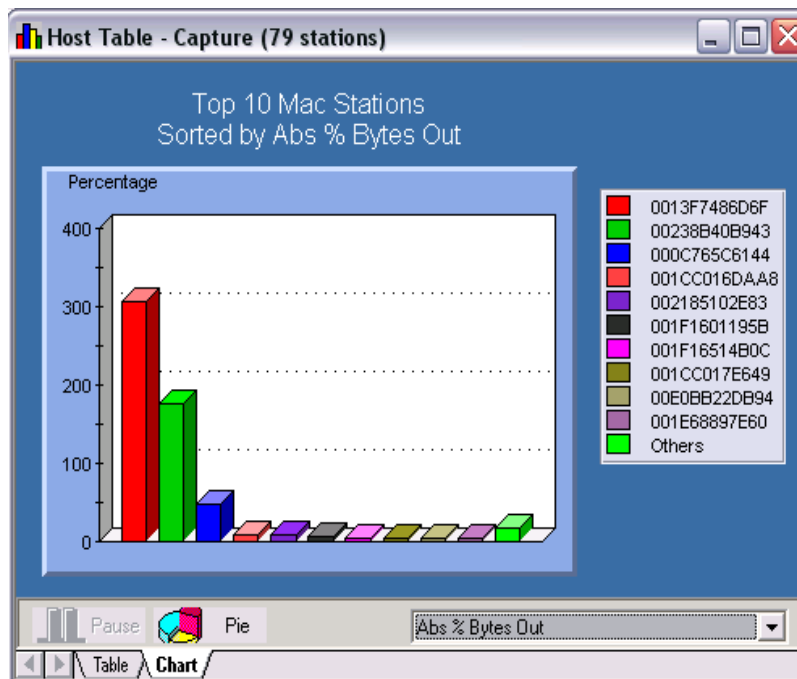
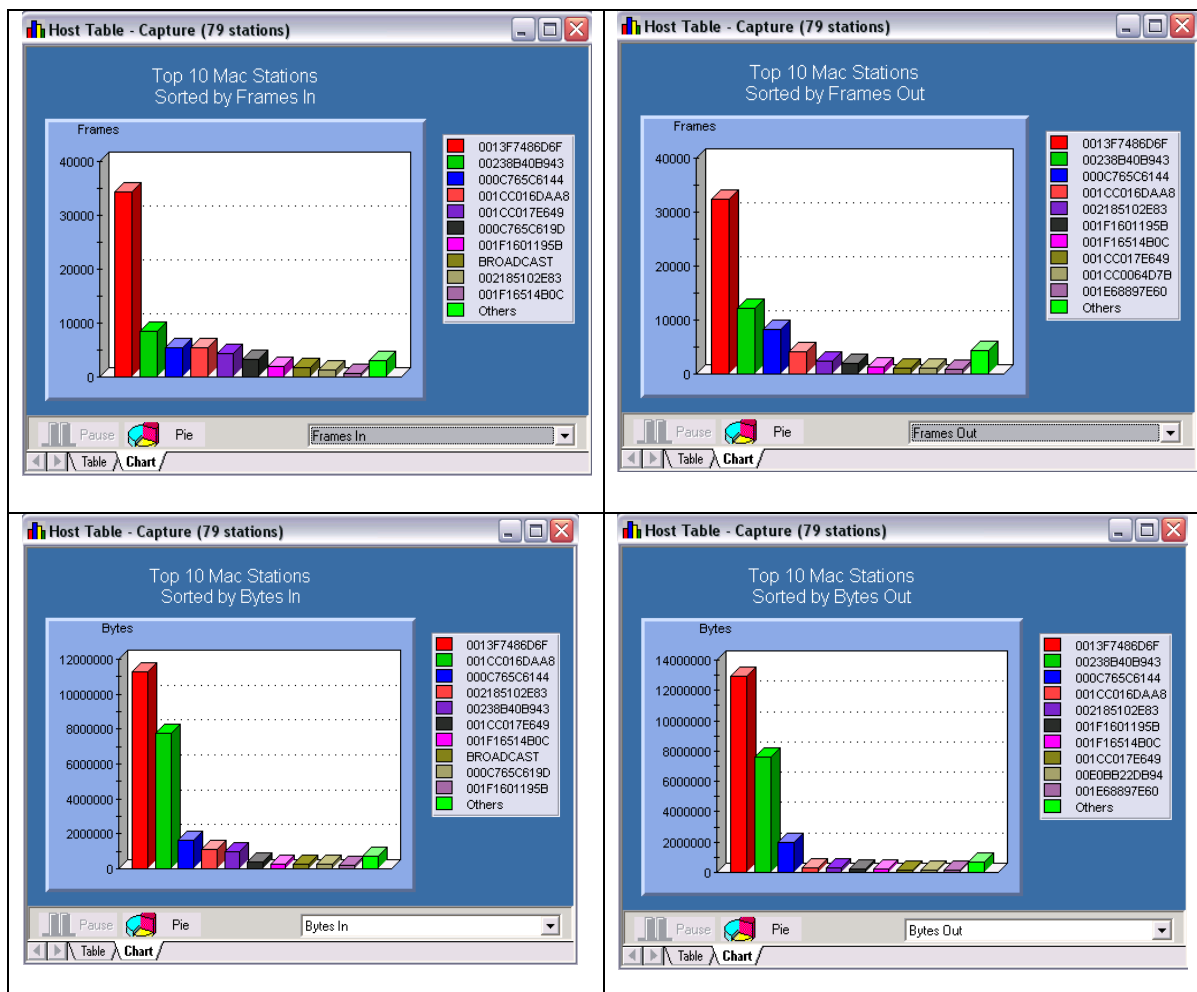


Figura. 3.21. Captura de paquetes según la Dirección MAC

3.1.2.5 Captura de Paquetes según la Dirección IP

Network Station Name	Frames In	Rel % Frames In	Frames Out	Bytes In	Rel % Bytes In	Bytes Out	Rel % Bytes Out
10.1.30.34	8439	12.04	12074	985652	3.97	7615560	30.70
10.1.0.112	5740	8.19	5650	1872446	7.54	1733227	6.98
10.1.30.212	5472	7.80	8158	1636898	6.60	2004999	8.08
10.1.30.197	5411	7.72	4109	7748705	31.23	293111	1.18
10.1.30.27	4265	6.08	1075	396099	1.59	172534	0.69
67.133.239.38	4082	5.82	5395	288788	1.16	7745963	31.22
121.254.90.65	3370	4.80	3335	3707067	14.94	215640	0.86
10.10.0.2	3207	4.57	377	241089	0.97	83406	0.33
kyo.deee.espe.edu.ec	3066	4.37	212	213586	0.86	30086	0.12
83.140.191.178	2599	3.70	0	176746	0.71	0	0.00
85.49.124.228	2578	3.67	1232	3276483	13.20	79947	0.32
10.1.30.26	1933	2.75	1974	213727	0.86	240257	0.96
10.1.30.19	1348	1.92	2398	1080939	4.35	294507	1.18
62.149.140.70	937	1.33	160	109160	0.44	58583	0.23
10.1.0.112	872	1.24	497	174203	0.70	227970	0.91
10.1.29.179	738	1.05	1254	263064	1.06	212332	0.85
80.190.154.63	644	0.91	0	46054	0.18	0	0.00
66.114.49.165	490	0.69	115	54768	0.22	64820	0.26
216.144.230.226	435	0.62	68	53117	0.21	28926	0.11
10.1.30.9	425	0.60	817	74558	0.30	143198	0.57
10.1.0.101	393	0.56	162	46847	0.18	26933	0.10
66.55.150.8	349	0.49	93	33066	0.13	24452	0.09
255.255.255.255	341	0.48	0	151844	0.61	0	0.00
174.133.179.117	267	0.38	117	43299	0.17	48821	0.19
66.115.190.84	250	0.35	48	24100	0.09	11472	0.04
85.214.45.22	241	0.34	95	29442	0.11	29842	0.12
208.22.33.46	240	0.34	348	17163	0.06	495022	1.99
67.15.150.143	245	0.34	71	40516	0.16	36620	0.14
10.1.30.22	222	0.31	575	58823	0.23	106542	0.42
85.116.134.41	221	0.31	67	42093	0.16	27443	0.11
10.1.30.1	213	0.30	369	18726	0.07	104351	0.42
10.1.0.134	198	0.28	115	22669	0.09	22037	0.08
74.125.159.118	183	0.26	226	19622	0.07	280378	1.13
66.55.150.7	163	0.23	57	20726	0.08	18867	0.07
10.1.30.255	162	0.23	0	22007	0.08	0	0.00
10.1.30.155	156	0.22	0	54600	0.22	0	0.00

Tabla. 3.13. Captura de paquetes según la Dirección IP

3.1.2.6 Numero de paquetes que circulan de acuerdo a la dirección IP

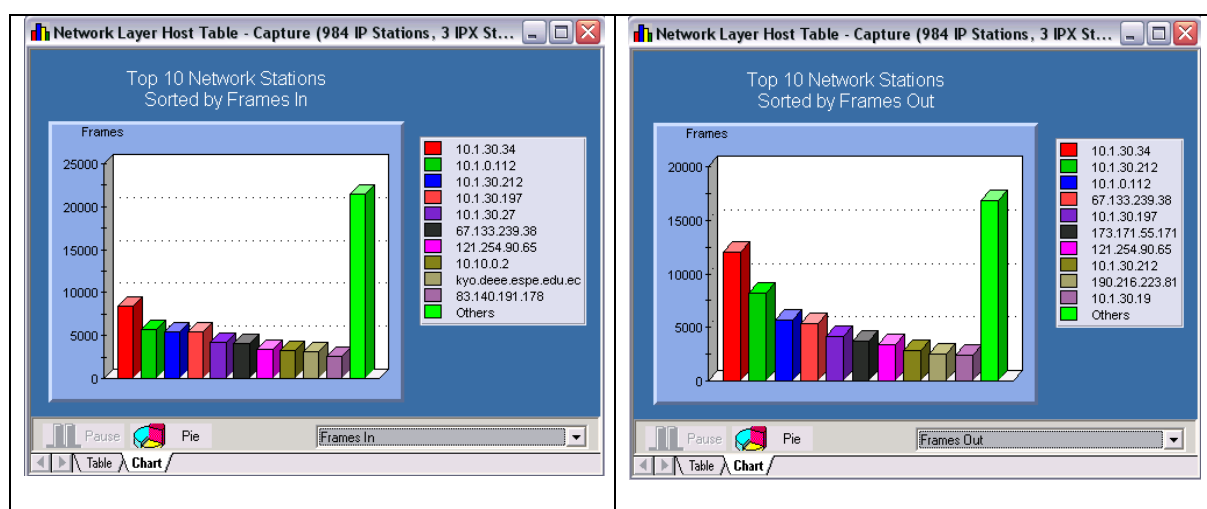


Figura. 3.22. Número de paquetes que circulan de acuerdo a la Dirección IP

3.1.2.7 Numero de bytes que circulan de acuerdo a la dirección IP

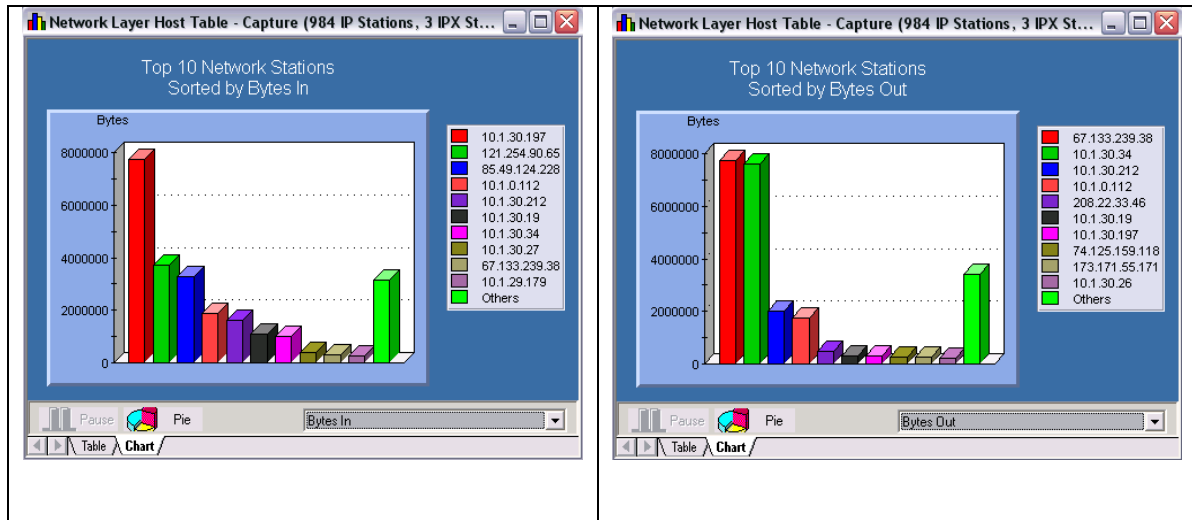


Figura. 3.23. Número de bytes que circulan de acuerdo a la Dirección IP

3.1.2.8 Tiempo de Respuesta

Server Name	Protocol	Minimum Time (ms)	Maximum Time (ms)	Average Time (ms)	Connections
10.1.0.101	DNS	0.001	135.032	5.916	266
10.1.0.104	DNS	0.001	116.803	16.254	40
10.1.30.1	DNS	0.001	170.716	19.662	174
10.1.30.212	CAST	0.001	0.404	0.136	12
10.1.30.9	DHCP/BOOTPSERV	0.004	19.131	0.456	154
10.10.0.1	DNS	0.002	31.168	3.126	26
10.10.0.2	IMAP	0.001	11.973	0.583	28
192.188.58.210	IMAP	0.004	12.375	0.615	28
207.246.136.196	HTTP	9.031	9.031	9.031	1
207.246.138.12	HTTP	9.024	9.024	9.024	1
207.246.138.140	TCP WKP: 81	4.334	13.691	9.016	3
207.246.153.238	HTTP	8.621	8.621	8.621	1
207.66.153.91	HTTP	76.615	76.615	76.615	1
208.49.52.82	HTTP	50.479	50.479	50.479	1
216.127.42.238	HTTP	13.695	13.695	13.695	1
65.54.50.228	MSNMS	0.387	1.721	1.054	2
74.125.45.83	HTTPS	0.839	0.845	0.842	4
74.125.67.113	HTTP	0.431	13.311	6.871	2
76.13.6.132	HTTP	0.523	0.523	0.523	1

Tabla. 3.14. Tiempo de respuesta

3.1.2.9 Resumen de toda la actividad generada por la interfaz eth1

➤ Capa de Transporte

Transport	
Symptoms	297
Analyses	2
Entities	696

➤ Symptoms:

Expert Symptom	Timestamp	Expert Summary
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP

Symptoms	
TCP Retransmission	4
TCP Fast Retransmission	249

Host 1: 10.1.0.112	
Packets In	5740
Packets Out	5650
Octets In	1872446
Octets Out	1733227
Non Unicast Packets Out	0
Associated MAC Address	0013F7:486D6F

Host 2: 10.1.30.212	
Packets In	5472
Packets Out	8158
Octets In	1638898
Octets Out	2004999
Non Unicast Packets Out	0
Associated MAC Address	000C76:5C6144

Tabla. 3.15. Resumen de toda la actividad generada por la interfaz eth1

➤ Entities

Client	Server	Protocol	First Frame	Last Frame	Analyses	Symptoms	Pkts In	Pkts Out	Octs In
10.1.0.112:8080	10.1.30.212:4011	TCP	2	5	0	0	2	2	40
10.1.0.112:8080	10.1.30.212:4013	TCP	6	49	0	1	22	22	6369
10.1.0.112:8080	10.1.30.212:4015	TCP	50	94	0	1	21	22	6349
10.1.0.112:8080	10.1.30.212:4017	TCP	95	142	0	1	21	21	6349
10.1.0.112:8080	10.1.30.212:4019	TCP	143	185	0	1	21	22	6349
10.1.0.112:8080	10.1.30.212:4021	TCP	187	229	0	1	21	22	6349
10.1.0.112:8080	10.1.30.212:4023	TCP	230	272	0	1	21	22	6349
10.1.0.112:8080	10.1.30.212:4025	TCP	273	317	0	1	21	22	6349
10.1.0.112:8080	10.1.30.212:4027	TCP	318	362	0	1	21	22	6349
10.1.0.112:8080	10.1.30.212:4029	TCP	365	415	0	1	21	22	6349

TCP: 8080 <-> 4011	
10.1.0.112	0013F7:486D6F
10.1.30.212	000C76:5C6144

10.1.0.112:8080 <-> 10.1.30.212:4011	
Packets In	2
Packets Out	2
Octets In	40
Octets Out	40
Start Time	Sat Dec 26 22:07:12
Stop Time	Sat Dec 26 22:07:12
Duration (s)	0

10.1.0.112:8080	
Max Window Size	65535
Min Window Size	65535
Retransmission	0
Zero Window Size	0
Analyses	0
Symptoms	0
Acknowledgments	1
Max Ack Time (ms)	0.002
Min Ack Time (ms)	0.002
Avg Ack Time (ms)	0.002

10.1.30.212:4011	
Max Window Size	65141
Min Window Size	65141
Retransmission	0
Zero Window Size	0
Analyses	0

Tabla. 3.16. Resumen de toda la actividad generada por la interfaz eth1

➤ Capa de Red

Network	
Symptoms	2000
Analyses	0
Entities	987

► Symptoms

Expert Symptom	Expert Summary
ICMP Time To Live Exceeded	Sent by Gateway [10.1.30.34] to [10.1.30.34] when forwarding to Destination [83.140.191.178], SA=[19
ICMP Time To Live Exceeded	Sent by Gateway [10.1.30.34] to [10.1.30.34] when forwarding to Destination [83.140.191.178], SA=[19
ICMP Time To Live Exceeded	Sent by Gateway [10.1.30.34] to [10.1.30.34] when forwarding to Destination [83.140.191.178], SA=[19
ICMP Time To Live Exceeded	Sent by Gateway [10.1.30.34] to [10.1.30.34] when forwarding to Destination [83.140.191.178], SA=[19
ICMP Time To Live Exceeded	Sent by Gateway [10.1.30.34] to [10.1.30.34] when forwarding to Destination [83.140.191.178], SA=[19
ICMP Time To Live Exceeded	Sent by Gateway [10.1.30.34] to [10.1.30.34] when forwarding to Destination [83.140.191.178], SA=[19
ICMP Time To Live Exceeded	Sent by Gateway [10.1.30.34] to [10.1.30.34] when forwarding to Destination [83.140.191.178], SA=[19
ICMP Time To Live Exceeded	Sent by Gateway [10.1.30.34] to [10.1.30.34] when forwarding to Destination [83.140.191.178], SA=[19
ICMP Time To Live Exceeded	Sent by Gateway [10.1.30.34] to [10.1.30.34] when forwarding to Destination [83.140.191.178], SA=[19
ICMP Time To Live Exceeded	Sent by Gateway [10.1.30.34] to [10.1.30.34] when forwarding to Destination [83.140.191.178], SA=[19

Symptoms	
ICMP Time To Live Exceeded	1941

Host 1: 190.216.223.81	
Packets In	0
Packets Out	2558
Octets In	0
Octets Out	199524
Non Unicast Packets Out	0
Associated MAC Address	0013F7:486D6F

Host 2: 10.1.30.34	
Packets In	8439
Packets Out	12074
Octets In	985652
Octets Out	7615560
Non Unicast Packets Out	38
Associated MAC Address	00238B:40B943

Tabla. 3.17. Resumen de toda la actividad generada por la interfaz eth1

➤ Entities

Network Station	Last MAC Station	Protocol	First Frame	Last Frame	Analyses	Symptoms	Pkts In	
10.1.30.1	0013F7:486D6F	IP	0	69743	0	3	213	
10.1.30.155	00507F:F05BC3	IP	0	69743	0	0	156	
10.1.30.212	000C76:5C6144	IP	2	68708	2	6	5472	
10.1.0.112	0013F7:486D6F	IP	2	70020	0	247	5740	
10.1.30.9	001E68:897E60	IP	62	70074	0	1	425	
255.255.255.255	FFFFFF:FFFFFF	IP	62	69748	0	0	341	
67.133.239.38	0013F7:486D6F	IP	90	70077	0	0	4082	
10.1.30.197	001CC0:16DAA8	IP	90	70077	0	0	5411	
10.1.30.34	00238B:40B943	IP	102	70072	1	4	8439	
83.140.191.178	0013F7:486D6F	IP	102	69998	0	0	2599	

Host: 10.1.30.1	
Packets In	213
Packets Out	369
Octets In	18726
Octets Out	104351
Non Unicast Packets Out	0
Associated MAC Address	0013F7:486D6F

Protocol					
Protocol	Frm <-	Frm ->	Bytes <-	Bytes ->	F. Frm
UDP	211	364	18514	103879	0
DNS	207	207	17960	48928	20936
DHCP/BOOTPSERV	1	157	350	54951	0
UDP OTHER	3	0	204	0	48895
ICMP	2	5	212	472	22300

Tabla. 3.18. Resumen de toda la actividad generada por la interfaz eth1

3.1.3 Interfaz eth2

3.1.3.1 Cantidad de Bytes por Protocolo

Protocol Name	Total Frames	Rel % Frames	Total Bytes	Rel % Bytes	Abs % Bytes
ARP	1645	2.34	109228	0.44	3.20
BPDU	207	0.29	11799	0.04	0.36
CAST	51	0.07	14479	0.05	0.35
CUPS	13	0.01	3159	0.01	0.08
DHCP/BOOTPSERV	479	0.68	206403	0.83	4.87
DHCPngS	10	0.01	1576	0.00	0.04
DNS	987	1.40	141779	0.57	3.64
DNS (TCP)	10	0.01	2440	0.00	0.06
ETHERNET II	69867	99.69	24792127	99.94	590.20
EtherType 0x8868	1055	1.50	284438	1.14	6.89
EtherType 0x88CC	14	0.01	2240	0.00	0.06
FTP	32	0.04	2945	0.01	0.08
HTTP	10243	14.61	2635707	10.62	64.01
HTTPS	401	0.57	102503	0.41	2.49
ICMP	5228	7.45	383119	1.54	10.99
IEEE 802.1D	207	0.29	11799	0.04	0.36
IEEE 802.1Q	69853	99.67	24789887	99.93	590.15
IEEE 802.2	258	0.36	17540	0.07	0.51
IGMP	2	0.00	136	0.00	0.00
IMAP	6425	9.16	456274	1.83	13.18
IP	67040	95.66	24367322	98.23	579.36
IPng	62	0.08	23158	0.09	0.55
IPX	10	0.01	794	0.00	0.02
IPX PROP	3	0.00	318	0.00	0.01
IPXNB	3	0.00	318	0.00	0.01
IPXRIP	7	0.00	476	0.00	0.01
KERBEROSU	2	0.00	2765	0.01	0.06
LLC SAP 0x5C	13	0.01	1489	0.00	0.04
LLC SAP 0x5E	21	0.02	2457	0.00	0.06
LLC SAP 0x78	7	0.00	1001	0.00	0.03
MGCP	8	0.01	1737	0.00	0.04
MMS	4	0.00	295	0.00	0.01
MSNMS	114	0.16	15589	0.06	0.40
NB-DATAGRAM	30	0.04	7361	0.02	0.18
NB-NAME	417	0.59	46668	0.18	1.24
NB-SESSION	73	0.10	11660	0.04	0.30
NBCIFS	364	0.51	57351	0.23	1.46

Tabla. 3.19. Cantidad de bytes por protocolo

3.1.3.2 Cantidad de Bytes

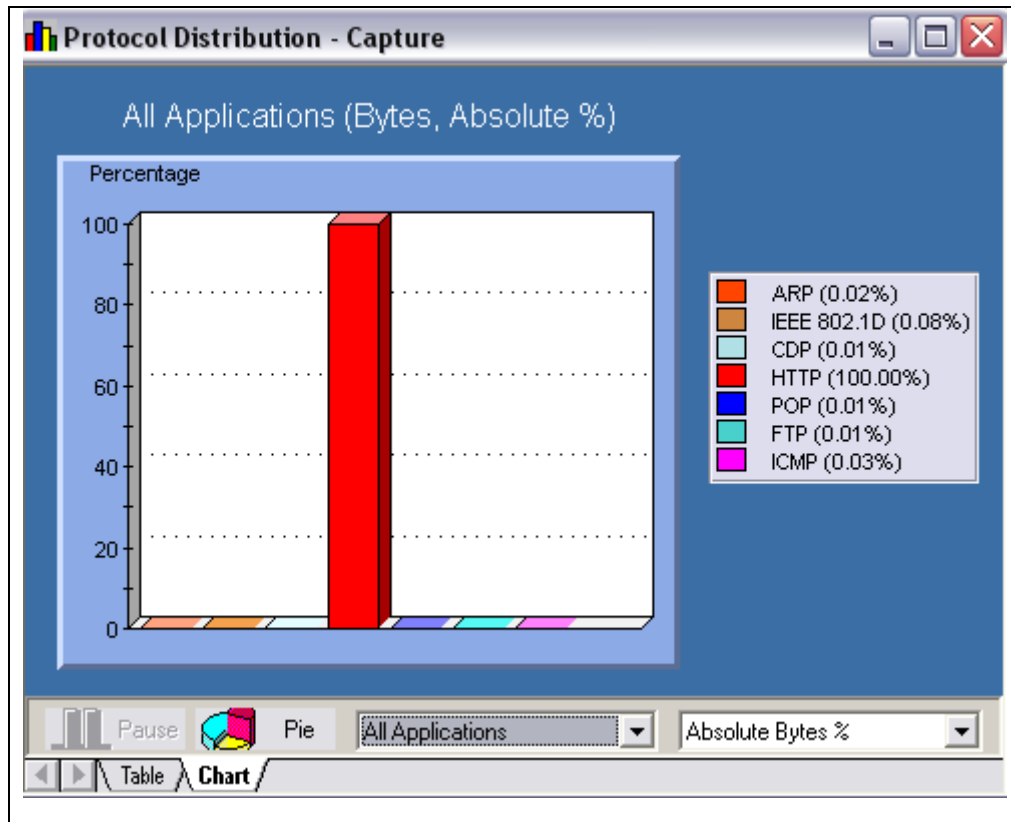


Figura. 3.24. Cantidad de bytes por protocolo

3.1.3.3 Distribución de Paquetes de acuerdo al tamaño

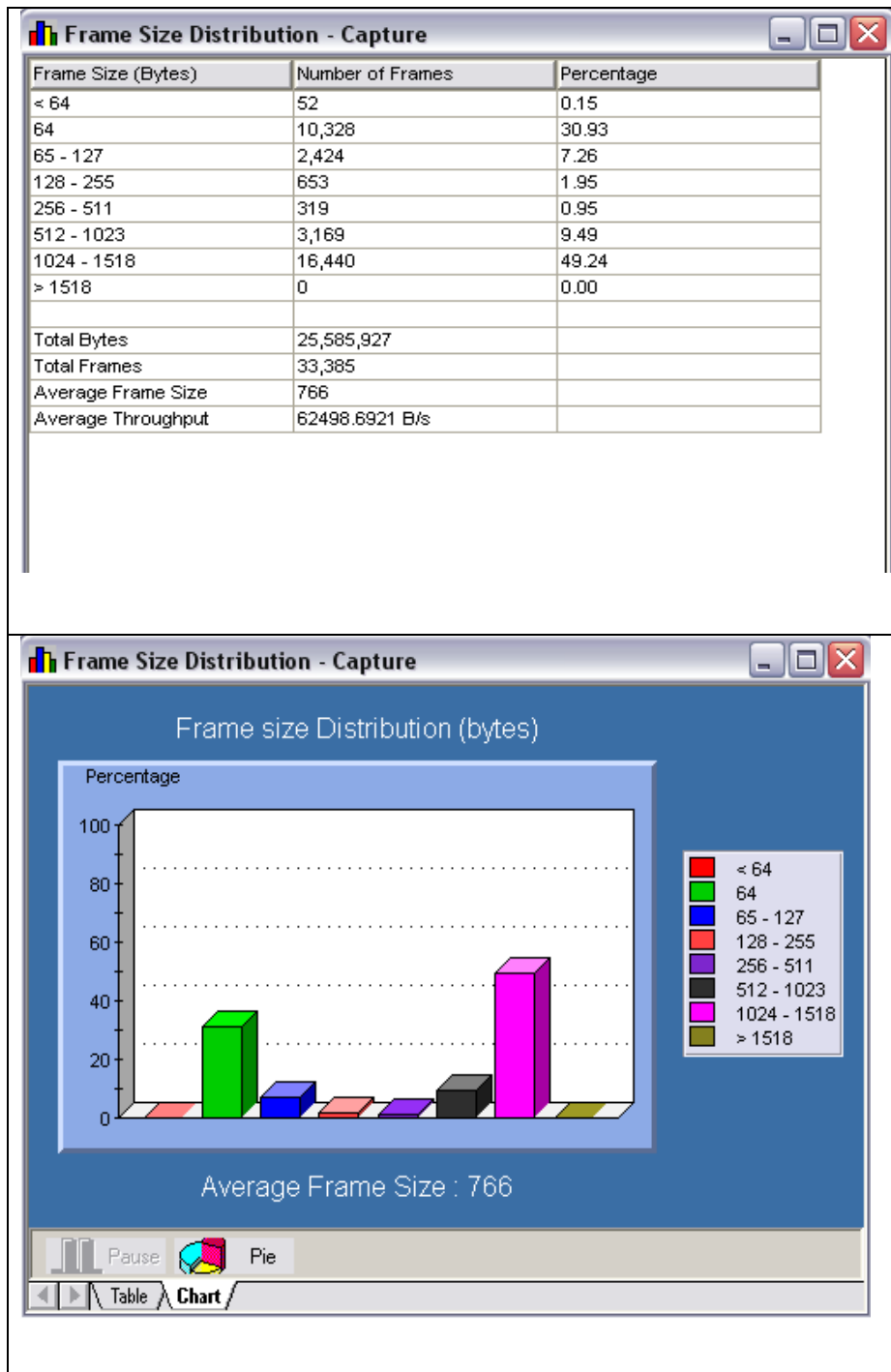


Figura 3.25. Distribución de paquetes de acuerdo al tamaño

3.1.3.4 Captura de Paquetes según la dirección MAC



Figura 3.26. Captura de paquetes según la dirección MAC

3.1.3.5 Captura de Paquetes según la Dirección IP

Network Station Name	Frames In	Rel % Frames In	Frames Out	Bytes In	Bytes Out
201.234.84.173	12760	38.22	20348	2956950	22611023
84.152.64.136	2704	8.09	1420	3559175	92292
123.240.72.175	2445	7.32	1213	3251995	78426
85.179.179.207	2197	6.58	1139	2857731	73588
113.103.25.239	1954	5.85	991	2573710	81008
94.195.152.12	1949	5.83	773	2584195	50762
92.102.53.115	1519	4.54	982	1669449	300364
131.114.242.179	1380	4.13	1417	1021989	613975
221.124.151.175	1022	3.06	600	1317109	64464
221.126.100.209	697	2.08	965	46902	560870
118.161.64.123	540	1.61	300	712989	20008
88.6.105.193	528	1.58	281	684613	18865
124.22.34.179	375	1.12	219	498328	14334
114.240.67.49	322	0.96	224	445033	15000
82.159.62.190	267	0.79	192	378728	12386
88.80.28.48	256	0.76	0	16384	0
82.154.92.54	249	0.74	328	16694	442793
201.233.21.162	192	0.57	95	245443	6120
71.178.189.5	148	0.44	206	9995	250483
116.27.217.199	147	0.44	89	183407	6318
98.156.121.222	123	0.36	56	128705	12126
115.132.234.59	84	0.25	38	117092	2474
68.82.182.87	73	0.21	55	74675	3520
81.245.59.178	47	0.14	27	52279	2174
77.204.1.87	40	0.11	33	37438	2112
123.118.9.235	36	0.10	49	2352	52198
137.224.236.57	25	0.07	30	1973	27287
123.6.81.95	20	0.05	21	1639	10318
110.44.0.50	19	0.05	0	1254	0
58.214.41.7	17	0.05	9	12557	1228
74.125.45.109	14	0.04	17	1358	3033
60.250.192.229	14	0.04	5	1279	490
113.128.67.19	13	0.03	12	1162	8371

Tabla. 3.20. Captura de paquetes según la dirección IP

3.1.3.6 Numero de paquetes que circulan de acuerdo a la dirección IP

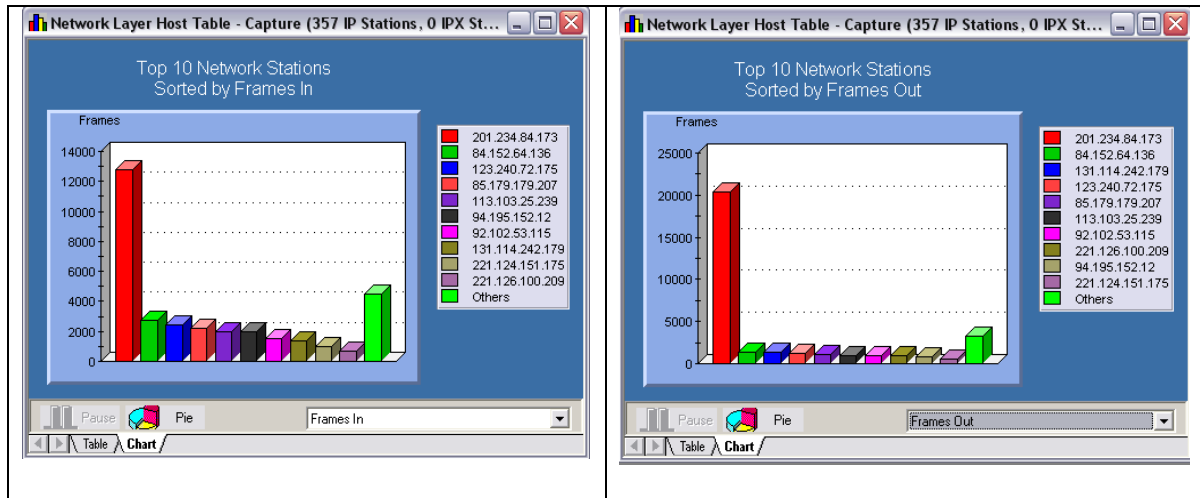


Figura. 3.27. Número de paquetes que circulan de acuerdo a la Dirección IP

3.1.3.7 Numero de bytes que circulan de acuerdo a la dirección IP

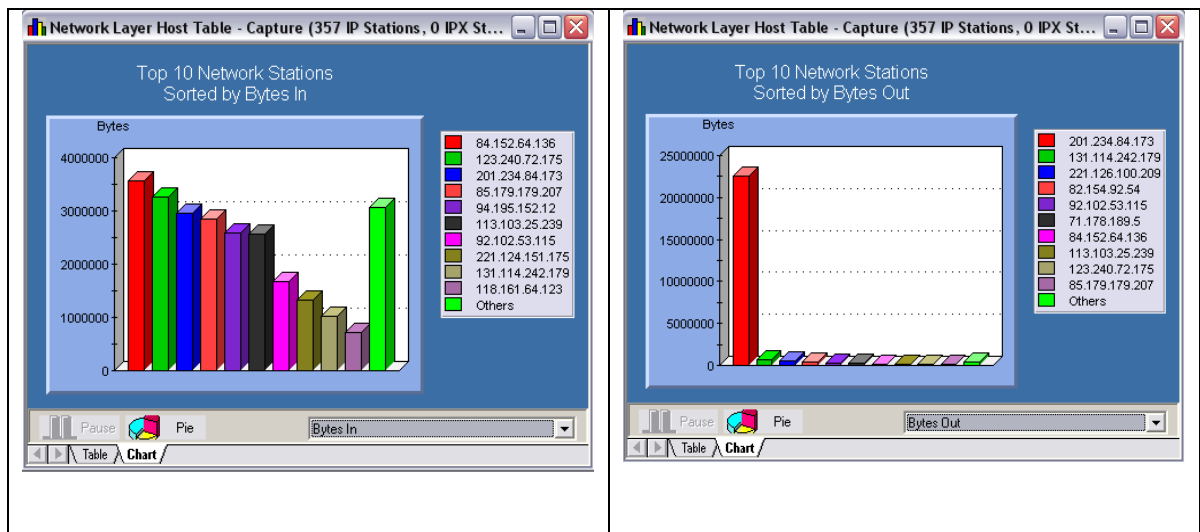


Figura. 3.28 Número de bytes que circulan de acuerdo a la Dirección IP

3.1.3.8 Tiempo de Respuesta

Server Name	Protocol	Minimum Time (ms)	Maximum Time (ms)	Average Time (ms)	Connections
200.115.33.83	POP3	0.004	17.686	6.741	3
200.82.187.6	TCP WKP: 68	28.419	364.184	231.937	6
201.234.84.173	KNETD	0.003	71.565	10.524	13
201.234.84.173	SSP	0.004	52.562	9.675	6
201.234.84.173	TDS	0.003	37.474	5.597	7
201.234.84.173	X.25	0.004	33.166	7.874	6
64.131.89.11	HTTP	74.666	74.666	74.666	1
65.54.48.167	MSNMS	19.544	24.490	22.017	2
65.54.48.53	MSNMS	27.688	43.445	35.566	2
65.54.48.94	MSNMS	17.001	44.997	31.549	4
65.54.50.179	MSNMS	23.988	31.544	26.588	3
65.54.93.101	HTTP	21.770	21.770	21.770	1
67.195.13.129	HTTP	160.246	160.246	160.246	1
67.220.205.164	HTTP	21.174	21.174	21.174	1
74.125.159.19	HTTPS	22.719	37.222	29.971	2
74.125.45.109	SSL	20.125	52.961	38.013	9
74.125.45.118	HTTP	49.764	49.764	49.764	1
74.125.45.19	HTTPS	53.022	53.475	53.174	3
84.111.27.151	TACACS+	202.384	337.122	267.253	6
88.6.105.193	TCP WKP: 85	29.337	324.913	93.280	521
89.185.229.182	HTTP	74.347	74.347	74.347	1
91.117.187.107	JABBER	44.621	65.277	54.949	2

Tabla. 3.21. Tiempo de respuesta

3.1.3.9 Resumen de toda la actividad generada por la interfaz eth2

➤ Capa de Transporte

Transport	
Symptoms	858
Analyses	7
Entities	145

➤ Symptoms:

Expert Symptom	Expert Summary
TCP Long Ack	Ack Time=[256 ms] between [94.195.152.12]/[TCP non-WKP: 4668] and [201.234.84.173]/[TCP non-WKP: 1948]
TCP Fast Retransmission	In 97 ms (< 100 ms) between [201.234.84.173]/[TCP non-WKP: 1948] and [94.195.152.12]/[TCP non-WKP: 4668]
TCP Retransmission	Between [201.234.84.173]/[TCP non-WKP: 1948] and [94.195.152.12]/[TCP non-WKP: 4668]
TCP Long Ack	Ack Time=[257 ms] between [201.234.84.173]/[TCP non-WKP: 4357] and [113.103.25.239]/[TCP non-WKP: 781]
TCP Long Ack	Ack Time=[234 ms] between [201.234.84.173]/[TCP non-WKP: 4357] and [113.103.25.239]/[TCP non-WKP: 781]
TCP Long Ack	Ack Time=[292 ms] between [124.22.34.179]/[TCP non-WKP: 7114] and [201.234.84.173]/[TCP non-WKP: 1760]
TCP Retransmission	Between [201.234.84.173]/[TCP non-WKP: 1760] and [124.22.34.179]/[TCP non-WKP: 7114]
TCP Retransmission	Between [201.234.84.173]/[TCP non-WKP: 1760] and [124.22.34.179]/[TCP non-WKP: 7114]
TCP Retransmission	Between [201.234.84.173]/[TCP non-WKP: 1760] and [124.22.34.179]/[TCP non-WKP: 7114]
TCP Fast Retransmission	In 64 ms (< 100 ms) between [221.126.100.209]/[TCP non-WKP: 4169] and [201.234.84.173]/[TCP non-WKP: 1760]

TCP Long Ack

- TCP: 4668 <-> 1948
 - 94.195.152.12
 - 001CF6:122E59
 - 201.234.84.173
 - 0013F7:486C47

Symptoms

Idle Too Long	2
TCP Retransmission	14
TCP Fast Retransmission	1
TCP Long Ack	17

Host 1: 94.195.152.12

Packets In	1949
Packets Out	773
Octets In	2584195
Octets Out	50762
Non Unicast Packets Out	0
Associated MAC Address	001CF6:122E59

Host 2: 201.234.84.173

Packets In	12760
Packets Out	20348
Octets In	2956950
Octets Out	22611023
Non Unicast Packets Out	0
Associated MAC Address	0013F7:486C47

Expert Symptom	Timestamp	Expert Summary
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP non-WKP: 4013]
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP non-WKP: 4013]
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP non-WKP: 4013]
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP non-WKP: 4013]
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP non-WKP: 4013]
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP non-WKP: 4013]
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP non-WKP: 4013]
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP non-WKP: 4013]
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP non-WKP: 4013]
TCP Fast Retransmission	Sat Dec 26 22:07:12	In 0 ms (< 100 ms) between [10.1.0.112]/[TCP non-WKP: 8080] and [10.1.30.212]/[TCP non-WKP: 4013]

TCP Fast Retransmission

- TCP: 8080 <-> 4013
 - 10.1.0.112
 - 0013F7:486D6F
 - 10.1.30.212
 - 000C76:5C6144

Symptoms

TCP Retransmission	4
TCP Fast Retransmission	249

Host 1: 10.1.0.112

Packets In	5740
Packets Out	5650
Octets In	1872446
Octets Out	1733227
Non Unicast Packets Out	0
Associated MAC Address	0013F7:486D6F

Host 2: 10.1.30.212

Packets In	5472
Packets Out	8158
Octets In	1638898
Octets Out	2004999
Non Unicast Packets Out	0
Associated MAC Address	000C76:5C6144

Tabla. 3.22. Resumen de toda la actividad generada por la interfaz eth2

➤ Analyses

Expert Analysis	Expert Summary	Frame ID	Ref Frame	Src Address	Dst Address
Too Many Retransmissions	Retransmission ratio is (4 / 16) = 25%	None	5835	201.234.84.173	115.132.234.59
Non Responsive Station	Station [60.250.192.229] not responding	17076	9722	201.234.84.173	60.250.192.229
Too Many Retransmissions	Retransmission ratio is (4 / 11) = 36%	None	17076	201.234.84.173	60.250.192.229
Non Responsive Station	Station [201.234.84.173] not responding	20129	16282	220.235.132.119	201.234.84.173
Non Responsive Station	Station [81.84.70.143] not responding	26167	15403	201.234.84.173	81.84.70.143
Too Many Retransmissions	Retransmission ratio is (8 / 33) = 24%	None	28017	201.234.84.173	81.245.59.178
Too Many Retransmissions	Retransmission ratio is (3 / 12) = 25%	None	31638	201.234.84.173	113.111.156.51

Too Many Retransmissions

- TCP: 7553 <-> 1781
- 201.234.84.173
- 115.132.234.59

Analyses

Too Many Retransmissions	1
--------------------------	---

Host 1: 201.234.84.173

Packets In	12760
Packets Out	20348
Octets In	2956950
Octets Out	22611023
Non Unicast Packets Out	0
Associated MAC Address	0013F7:486C47

Host 2: 115.132.234.59

Packets In	84
Packets Out	38
Octets In	117092
Octets Out	2474
Non Unicast Packets Out	0
Associated MAC Address	001CF6:122E59

Tabla 3.23. Resumen de toda la actividad generada por la interfaz eth2

➤ Entities

Client	Server	Protocol	First Frame	Last Frame	Analyses	Symptoms	Pkts In	Pkts Out	Octs
24.245.60.176:4661	201.234.84.173:1963	TCP	1390	1442	0	0	2	2	40
62.99.124.79:4662	201.234.84.173:1962	TCP	1389	1474	0	0	2	2	40
79.86.182.218:15210	201.234.84.173:1964	TCP	1391	1481	0	0	2	2	40
83.35.225.229:50131	201.234.84.173:1961	TCP	1388	1468	0	0	2	2	40
95.114.247.100:4662	201.234.84.173:1965	TCP	1804	2593	0	1	3	2	60
117.32.189.104:6256	201.234.84.173:1968	TCP	2384	2496	0	0	2	2	40
151.21.20.46:4662	201.234.84.173:1966	TCP	2304	2376	0	0	2	2	40
201.234.84.173:3173	65.54.93.101:80	TCP	3154	4231	0	0	4	6	254
93.97.252.42:7951	201.234.84.173:1974	TCP	4336	4408	0	0	2	2	40
114.89.0.78:5144	201.234.84.173:1972	TCP	4847	5076	0	0	2	2	40

TCP: 4661 <-> 1963	
24.245.60.176	001CF6:122E59
201.234.84.173	0013F7:486C47

24.245.60.176:4661 <-> 201.234.84.173:1963	
Packets In	2
Packets Out	2
Octets In	40
Octets Out	40
Start Time	Sat Feb 06 12:40:15
Stop Time	Sat Feb 06 12:40:15
Duration (s)	0

24.245.60.176:4661	
Max Window Size	65258
Min Window Size	65258
Retransmission	0
Zero Window Size	0
Analyses	0
Symptoms	0
Acknowledgments	1
Max Ack Time (ms)	32.182
Min Ack Time (ms)	32.182
Avg Ack Time (ms)	32.182

201.234.84.173:1963	
Max Window Size	65329
Min Window Size	65329
Retransmission	0
Zero Window Size	0

Tabla 3.24. Resumen de toda la actividad generada por la interfaz eth2

➤ Capa de Red

Network	
Symptoms	250
Analyses	0
Entities	357

► Symptoms

Expert Symptom	Expert Summary
ICMP Time To Live Exceeded	Sent by Gateway [201.234.84.173] to [201.234.84.173] when forwarding to Destination [88.80.28.48]. SA:
ICMP Time To Live Exceeded	Sent by Gateway [201.234.84.173] to [201.234.84.173] when forwarding to Destination [88.80.28.48]. SA:
ICMP Time To Live Exceeded	Sent by Gateway [201.234.84.173] to [201.234.84.173] when forwarding to Destination [88.80.28.48]. SA:
ICMP Time To Live Exceeded	Sent by Gateway [201.234.84.173] to [201.234.84.173] when forwarding to Destination [88.80.28.48]. SA:
ICMP Time To Live Exceeded	Sent by Gateway [201.234.84.173] to [201.234.84.173] when forwarding to Destination [88.80.28.48]. SA:
ICMP Time To Live Exceeded	Sent by Gateway [201.234.84.173] to [201.234.84.173] when forwarding to Destination [88.80.28.48]. SA:
ICMP Time To Live Exceeded	Sent by Gateway [201.234.84.173] to [201.234.84.173] when forwarding to Destination [88.80.28.48]. SA:
ICMP Time To Live Exceeded	Sent by Gateway [201.234.84.173] to [201.234.84.173] when forwarding to Destination [88.80.28.48]. SA:
ICMP Time To Live Exceeded	Sent by Gateway [201.234.84.173] to [201.234.84.173] when forwarding to Destination [88.80.28.48]. SA:
ICMP Time To Live Exceeded	Sent by Gateway [201.234.84.173] to [201.234.84.173] when forwarding to Destination [88.80.28.48]. SA:

ICMP Time To Live Exceeded	
190.216.223.81	
001CF6:122E59	
201.234.84.173	
0013F7:486C47	

Symptoms	
ICMP Host Unreachable	13
ICMP Time To Live Exceeded	229

Host 1: 190.216.223.81	
Packets In	0
Packets Out	242
Octets In	0
Octets Out	17908
Non Unicast Packets Out	0
Associated MAC Address	001CF6:122E59

Host 2: 201.234.84.173	
Packets In	12760
Packets Out	20348
Octets In	2956950
Octets Out	22611023
Non Unicast Packets Out	0
Associated MAC Address	0013F7:486C47

Tabla 3.25. Resumen de toda la actividad generada por la interfaz eth2

► Entities

Network Station	Last MAC Station	Protocol	First Frame	Last Frame	Analyses	Symptoms	Pkts In	Pkts Out	Octs In	Octs O
94.195.152.12	001CF6:122E59	IP	0	33024	0	17	1949	773	2584195	50762
201.234.84.173	0013F7:486C47	IP	0	33174	6	392	12760	20348	2956950	226110
85.179.179.207	001CF6:122E59	IP	4	33172	0	1	2197	1139	2857731	73588
113.103.25.239	001CF6:122E59	IP	7	33119	0	3	1954	991	2573710	81008
221.126.100.209	001CF6:122E59	IP	11	33168	0	26	697	965	46902	560870
84.127.62.134	001CF6:122E59	IP	19	13211	0	1	7	7	669	761
84.152.64.136	001CF6:122E59	IP	20	33131	0	3	2704	1420	3559175	92292
124.22.34.179	001CF6:122E59	IP	34	32991	0	182	375	219	498328	14334
92.102.53.115	001CF6:122E59	IP	40	33146	0	14	1519	982	1669449	30036
123.240.72.175	001CF6:122E59	IP	48	33110	0	2	2445	1213	3251995	78426

Host: 94.195.152.12	
Packets In	1949
Packets Out	773
Octets In	2584195
Octets Out	50762
Non Unicast Packets Out	0
Associated MAC Address	001CF6:122E59

Protocol						
Protocol	Frm <-	Frm ->	Bytes <-	Bytes ->	F. Frm	L. Frm
TCP	1949	773	2584195	50762	0	33024
TCP OTHER	1949	773	2584195	50762	0	33024

Tabla 3.26. Resumen de toda la actividad generada por la interfaz eth2

CAPITULO 4

ANÁLISIS DEL TRÁFICO DE RED TCP/IP SOBRE UNA RED DE PRUEBAS

1. IMPLEMENTACIÓN DE LA RED

La red de pruebas implementada (figura 3.1), la cual se utilizará para simular los diferentes tipos de ataque, es la que se muestra a continuación:

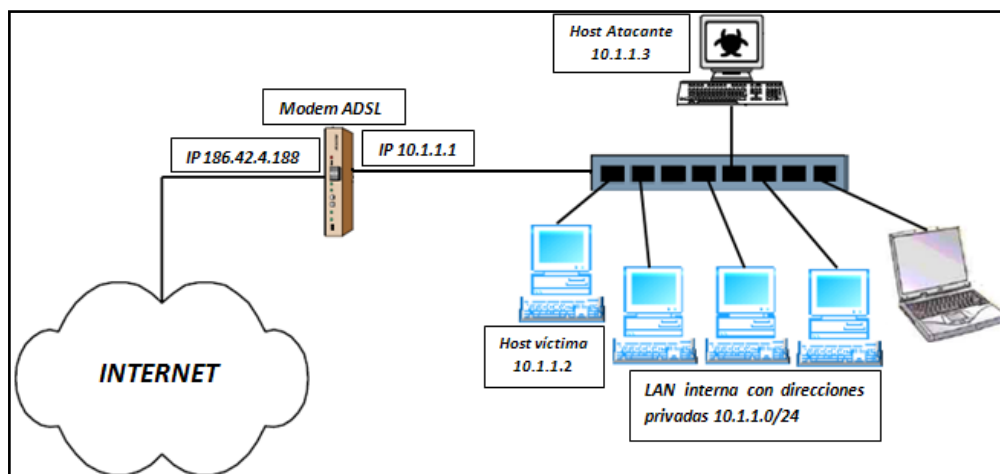


Figura. 4.1. Red de Pruebas Implementada

Se ha utilizado una red interna cuya dirección utilizada es la 10.1.1.0/24, esta red posee conexión a internet a través de un modem ADSL.

Dadas las facilidades que hoy en día presentan los sistemas operativos de código abierto, en la implementación y ejecución de herramientas que permitan explotar las vulnerabilidades que presentan las capas del modelo

TCP/IP, el host atacante está instalado con el sistema operativo LINUX (CENTOS 5.9).

Los hosts que estarán involucrados directa o indirectamente en los ataques a implementarse, que pertenecen a la red interna están operando con el sistema operativo Windows.

1.1 Herramientas utilizadas

1.1.1 NMAP (Network Mapper).

Es una herramienta que permite la exploración y la auditoria de redes de ordenadores. Su misión principal consiste en la realización de varios tipos de exámenes de puertos (*port scanning*) para la obtención de los servicios existentes en un ordenador. La versión que se ha utilizado ha sido la 3.3.0.

Este software gratuito, muy estable que funciona tanto en sistemas Unix como Windows. Es ampliamente conocido y utilizado por administradores de red y *hackers*.

El objetivo del uso de esta herramienta es la de verificar los servicios accesibles que dispone el host víctima dentro de la red.

Otra característica de NMAP es la de efectuar la detección del sistema operativo existente en el ordenador (*OS fingerprinting*). Así como también auditar al host víctima y obtener información relevante.

La instalación efectuada es la que se recomienda por defecto. Sin embargo cabe notar que para ejecutar todas las funcionalidades que proporciona esta herramienta se debe tener privilegios de administrador (*root*).

2. ACTIVIDADES Y MÉTODOS MÁS COMUNES DE ATAQUE

2.1 Actividades previas a la realización de un ataque

Previo a la planificación y ejecución de un posible ataque contra uno o más equipos de una red TCP/IP, es necesario conocer el objetivo que hay que atacar. Para realizar esta actividad, es decir, para obtener toda la información posible de la víctima, será necesario utilizar una serie de técnicas y herramientas de obtención y recolección de información.

A continuación se presentan, algunas de las técnicas existentes que tanto los administradores de una red como los posibles atacantes, pueden utilizar para obtener información del host objetivo.

2.1.1 Utilización de herramientas de administración.

La utilización de todas aquellas herramientas de administración: *ping*, *traceroute*, *whois*, *finger*, *rusers*, *nslookup*, *rcpinfo*, *telnet*, *dig*, etc permiten la obtención de información de un sistema.

2.1.1.1 La herramienta Ping

La ejecución del comando ***ping*** a una dirección IP asociada a un nombre de dominio podría ofrecer al atacante información de gran utilidad pudiendo determinar la existencia de uno o más equipos conectados a la red de este dominio.

La salida del comando *ping* permite conocer:

- La dirección IP que corresponde al nombre de la máquina remota.

- El número de secuencia ICMP.
- La vida útil del paquete (*TTL*).
- El campo de demora de vueltas corresponde al lapso de tiempo en milisegundos que se necesita para dar una vuelta entre las máquinas fuente y destino. Como regla general, la demora de un paquete no debe ser mayor a 200 ms.
- La cantidad de paquetes perdidos.

2.1.1.2 Traceroute.

Esta herramienta permite determinar la ruta efectuada por un paquete. El comando *traceroute* se puede usar para diagramar un mapa de los routers que se encontraron entre la máquina fuente y la máquina destino.

Una vez descubierta la existencia de como mínimo, uno de los equipos del dominio, el atacante podría ejecutar el comando *traceroute* para obtener información relacionada con la topología o la distribución física y lógica de la red.

Aunque *traceroute* es una herramienta de administración pensada para solucionar problemas de red, también se puede utilizar con objetivos deshonestos. Por ejemplo, *traceroute* se puede emplear para tratar de averiguar que sistemas existen entre distintos equipos.

El funcionamiento de *traceroute* se basa en la manipulación del campo *TTL* de la cabecera IP de un paquete, de forma que es capaz de determinar uno a uno los saltos por los que un determinado paquete avanza por la red TCP/IP.

2.1.1.3 Netstat.

Netstat es una herramienta que permite identificar las conexiones TCP que están activas en la máquina en la que se ejecuta el comando. A su vez, esta

herramienta crea una lista con todos los puertos TCP y UDP que están abiertos en el ordenador.

El comando "netstat" también permite a su vez obtener estadísticas de numerosos protocolos (Ethernet, IPv4, TCP, UDP, ICMP y IPv6).

Cuando se lo utiliza sin argumentos, el comando **netstat** muestra todas las conexiones abiertas por el ordenador. El comando **netstat** posee una serie de configuraciones opcionales, cuya sintaxis es la siguiente: La sintaxis es:

netstat [-a] [-e] [-n] [-o] [-s] [-p PROTO] [-r] [interval]

Cuando se utiliza con el argumento *-a*, el comando **netstat** muestra todas las conexiones y los puertos en escucha de la máquina.

Cuando se lo utiliza con el argumento *-e*, el comando **netstat** muestra las estadísticas Ethernet.

Cuando se lo utiliza con el argumento *-n*, el comando **netstat** muestra las direcciones y los números de puerto en forma numérica, sin resolución de nombres.

Cuando se lo utiliza con el argumento *-o*, el comando **netstat** indica el número del proceso asignado a la conexión.

Cuando se lo utiliza con el argumento *-p* seguido del nombre del protocolo (TCP, UPD o IP), el comando **netstat** muestra la información solicitada relacionada con el protocolo especificado.

Cuando se lo utiliza con el argumento *-r*, el comando **netstat** muestra la tabla de enrutamiento.

Cuando se lo utiliza con el argumento `-s`, el comando **netstat** muestra las estadísticas detalladas para cada protocolo.

Por último, un intervalo opcional permite determinar el período de actualización de la información, en segundos. El tiempo predeterminado es de 1 segundo.

2.1.1.4 Whois

La herramienta "Whois" tiene dos propósitos fundamentales:

- Obtener información con respecto al dueño de un nombre de dominio (con propósitos administrativos, técnicos y posiblemente de facturación) y los servidores asociados con el dominio.
- Obtener información con respecto al dueño de una cantidad determinada de direcciones IP.

2.1.1.5 Nslookup

Nslookup (*Name System Lookup*). Es una herramienta que permite consultar un servidor de nombres y obtener información relacionada con el dominio o el host.

El atacante tratará de obtener toda aquella información general relacionada con la organización que hay detrás de la red que se quiere atacar. La recogida de esta información puede empezar extrayendo la información relativa a los dominios asociados a la organización, así como las subredes correspondientes.

Esto puede obtenerse fácilmente mediante consultas al servicio de nombre de dominios (*DNS*).

Es posible solicitar información sobre un host indicando su nombre seguido del comando *nslookup*:

nslookup nombre.del.host

De modo predeterminado, el comando *nslookup* realiza consultas al servidor de nombres primario configurado en la máquina. Sin embargo, es posible consultar un servidor de nombres específico, agregando un signo menos después del comando:

nslookup nombre.del.host -nombre.del.servidor

Es posible modificar el modo de consulta del comando *nslookup* usando el argumento *set*:

- **set type=mx**: Permite obtener información relacionada con el(los) servidor(es) de correo de un dominio.
- **set type=ns**: Permite obtener información del servidor de nombres relacionado al dominio.
- **set type=a**: Permite obtener información de un host de la red. Se trata de un modo de consulta predeterminado.
- **set type=soa**: Permite mostrar la información del campo *SOA* (*inicio de autoridad*).
- **set type=cname**: Permite mostrar información relacionada con los alias.
- **set type=info**: Permite mostrar, siempre y cuando los datos estén disponibles, la información relacionada con el material y el sistema operativo del host.

Entre la información encontrada, el atacante podría encontrar información sensible como, por ejemplo, relaciones entre sistemas de la red o subredes, el

objetivo para el que se utilizan los mismos, el sistema operativo instalado en cada equipo, etc.

2.1.2 Exploración de puertos.

La exploración de puertos es una técnica utilizada para identificar los servicios que ofrecen los sistemas de destino. Suele ser la última de las actividades previas a la realización de un ataque.

Al explorar los puertos de los sistemas se descubren puntos de entrada a los mismos, que abren las puertas a nuevas vulnerabilidades potenciales en base a la implementación del servidor que escucha tras cada puerto. Además, esta técnica también permite identificar el tipo de sistema existente, su sistema operativo, y las aplicaciones que ofrecen un servicio en la red, así como su versión asociada.

2.1.2.1 Exploración de puertos TCP.

La exploración de puertos TCP se puede utilizar para descubrir si dicho sistema ofrece o no un determinado servicio.

Las técnicas existentes para realizar exploración de puertos TCP emplean diferentes procedimientos para descubrir la información del servicio:

- **TCP connect scan.** Mediante el establecimiento de una conexión TCP (completando los tres pasos del establecimiento de la conexión) la exploración puede ir analizando todos los puertos posibles. Si la conexión se realiza correctamente, se anotará el puerto como abierto (realizando una suposición de su servicio asociado según el número de puerto).

- **TCP SYN scan.** Enviando únicamente paquetes de inicio de conexión (SYN) por cada uno de los puertos que se quieren analizar se puede determinar si estos están abiertos o no. Recibir como respuesta un paquete RST-ACK significa que no existe ningún servicio que escuche por este puerto.

Por el contrario, si se recibe un paquete SYN-ACK, podemos afirmar la existencia de un servicio asociado a dicho puerto TCP. En este caso, se enviará un paquete RST-ACK para no establecer conexión y no ser registrados por el sistema objetivo, a diferencia del caso anterior (*TCP connect scan*).

- **TCP FIN scan.** Al enviar un paquete FIN a un puerto, deberíamos recibir un paquete de reset (RST) si dicho puerto está cerrado. Esta técnica se aplica principalmente sobre implementaciones de pilas TCP/IP de sistemas Unix.
- **TCP Xmas Tree scan.** Esta técnica es muy similar a la anterior, y también se obtiene como resultado un paquete de reset si el puerto está cerrado. En este caso se envían paquetes FIN, URG y PUSH.
- **TCP Null scan.** En el caso de poner a cero todos los indicadores de la cabecera TCP, la exploración debería recibir como resultado un paquete de reset en los puertos no activos.

2.1.2.2 Exploración de puertos UDP.

Mediante la exploración de puertos UDP es posible determinar si un sistema está o no disponible, así como encontrar los servicios asociados a los puertos UDP que encontramos abiertos.

Dado que UDP es un protocolo no orientado a conexión, la fiabilidad de este método depende de numerosos factores (más todavía en internet), como

son la utilización de la red y sus recursos, la carga existente, la existencia de filtros de paquetes en sistemas finales o en sistemas cortafuegos, etc.

Asimismo, y a diferencia de las exploraciones TCP, se trata de un proceso mucho más lento, puesto que la recepción de los paquetes enviados se consigue mediante el vencimiento de temporizadores (*timeouts*).

2.1.2.3 Herramientas para realizar la exploración de puertos.

La herramienta que permite realizar exploración de puertos es *Nmap* (*Network Mapper*), pues implementa la gran mayoría de técnicas conocidas para exploración de puertos y permite descubrir información de los servicios y sistemas encontrados.

Mediante *Nmap* pueden realizarse, las siguientes acciones de exploración:

- Exploración de direcciones IP activas.
- **nmap -sP IP ADDRESS/NETMASK**
- Exploración de puertos TCP activos:
- **nmap -sT IP ADDRESS/NETMASK**
- Exploración de puertos UDP activos:
- **nmap -sU IP ADDRESS/NETMASK**
- Exploración del tipo de sistema operativo de un equipo en red:
- **nmap -O IP ADDRESS/NETMASK**

2.1.2.4 Análisis de resultados.

La mayoría de herramientas de exploración de puertos dejan evidencias en los hosts destino, del intento de obtener información de manera ilícita. Los administradores de red pueden percatarse de dicha exploración monitoreando el tráfico que circula por la red, así por ejemplo empleando la herramienta

wireshark en una exploración de puertos TCP activos solicitada desde el host 10.1.1.3 hacia toda la red y monitoreando el tráfico que circula por la interfaz de red del host 10.1.1.2 tenemos lo siguiente:

El host 10.1.1.3 ha ejecutado el siguiente comando:

```
[root@localhost ~]# nmap -sS 10.1.1.0/24
```

```
Starting Nmap 5.00 (http://nmap.org ) at 2009-10-15 17:10 ECT
```

```
Interesting ports on 10.1.1.1:
```

```
Not shown: 997 closed ports
```

```
PORT      STATE SERVICE
```

```
21/tcp    open  ftp
```

```
23/tcp    open  telnet
```

```
80/tcp    open  http
```

```
MAC Address: 00:1E:58:92:78:4B (D-Link)
```

```
Interesting ports on 10.1.1.2:
```

```
Not shown: 994 closed ports
```

```
PORT      STATE SERVICE
```

```
135/tcp   open  msrpc
```

```
1110/tcp  open  nfsd-status
```

```
3476/tcp  open  unknown
```

```
3580/tcp  open  unknown
```

```
19780/tcp open  unknown
```

```
61900/tcp open  unknown
```

```
MAC Address: 00:23:8B:7A:21:96 (Quanta Computer)
```

```
All 1000 scanned ports on 10.1.1.3 are filtered
```

```
MAC Address: 00:16:B6:2C:EA:40 (Cisco-Linksys)
```

```
Interesting ports on 10.1.1.4:
```

```
Not shown: 998 closed ports
```

```
PORT      STATE SERVICE
```

```
22/tcp    open  ssh
```

111/tcp open rpcbind

Nmap done: 256 IP addresses (4 hosts up) scanned in 19.22 seconds

Los resultados son evidentes, muestran los puertos TCP abiertos, en cada uno de los host activos de la subred 10.1.1.10/24.

Todas aquellas opciones de Nmap precedidas por el prefijo -s son consideradas exploraciones silenciosas (pero detectables al igual que cualquier exploración de puertos). Analizando el tráfico que circula por la interfaz del host 10.1.1.2 al momento de la exploración se tiene lo siguiente:

No. -	Time	Source	Destination	Protocol	Info
4	0.202924	QuantaCo_7a:21:96	EdimaxTe_88:80:fd	ARP	10.1.1.2 is at 00:23:8b:7a:21:96
5	0.202949	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.3? Tell 10.1.1.4
6	0.202957	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.0? Tell 10.1.1.4
7	0.303718	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.0? Tell 10.1.1.4
8	0.303737	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.3? Tell 10.1.1.4
9	0.417608	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.1? Tell 10.1.1.4

Tabla. 4.1. Tráfico que circula por la interfaz del host 10.1.1.2

Como primer paso Nmap, con la ayuda de la dirección de broadcast y del protocolo ARP, verifica los hosts activos en la subred. Para la exploración de puertos TCP se basa en el envío de paquetes TCP/SYN. Si bien Nmap no finaliza el protocolo de conexión de forma expresa, es capaz de recoger la información suficiente para detectar todos aquellos servicios TCP ofrecidos por los hosts explorados, la siguiente gráfica ilustra lo descrito anteriormente.

No. -	Time	Source	Destination	Protocol	Info
21	4.476061	10.1.1.4	10.1.1.2	TCP	57825 > pop3s [SYN] Seq=0 win=5840 Len=0 MSS=1460 TSV=
22	4.476069	10.1.1.2	10.1.1.4	TCP	pop3s > 57825 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
23	4.476092	10.1.1.4	10.1.1.2	TCP	45935 > pptp [SYN] Seq=0 win=5840 Len=0 MSS=1460 TSV=
24	4.476101	10.1.1.2	10.1.1.4	TCP	pptp > 45935 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
25	4.476122	10.1.1.4	10.1.1.2	TCP	60588 > smux [SYN] Seq=0 win=5840 Len=0 MSS=1460 TSV=
26	4.476131	10.1.1.2	10.1.1.4	TCP	smux > 60588 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
27	4.476438	10.1.1.4	10.1.1.2	TCP	55447 > blackjack [SYN] Seq=0 win=5840 Len=0 MSS=1460
28	4.476449	10.1.1.2	10.1.1.4	TCP	blackjack > 55447 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
29	4.476474	10.1.1.4	10.1.1.2	TCP	45059 > dd1-tcp-1 [SYN] Seq=0 win=5840 Len=0 MSS=1460
30	4.476482	10.1.1.2	10.1.1.4	TCP	dd1-tcp-1 > 45059 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
31	4.476867	10.1.1.4	10.1.1.2	TCP	41639 > http [SYN] Seq=0 win=5840 Len=0 MSS=1460 TSV=
32	4.476876	10.1.1.2	10.1.1.4	TCP	http > 41639 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
33	4.476897	10.1.1.4	10.1.1.2	TCP	57940 > vnc-server [SYN] Seq=0 win=5840 Len=0 MSS=1460
34	4.476906	10.1.1.2	10.1.1.4	TCP	vnc-server > 57940 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
35	4.476926	10.1.1.4	10.1.1.2	TCP	39084 > microsoft-ds [SYN] Seq=0 win=5840 Len=0 MSS=1460
36	4.476934	10.1.1.2	10.1.1.4	TCP	microsoft-ds > 39084 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
37	4.476960	10.1.1.4	10.1.1.2	TCP	43475 > https [SYN] Seq=0 win=5840 Len=0 MSS=1460 TSV=
38	4.476968	10.1.1.2	10.1.1.4	TCP	https > 43475 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Tabla. 4.2. Hosts activos en la subred

Como último paso Nmap verifica que no exista más hosts activos en la red.

No. -	Time	Source	Destination	Protocol	Info
2044	5.911994	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.7? Tell 10.1.1.4
2045	5.912003	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.8? Tell 10.1.1.4
2046	5.912450	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.9? Tell 10.1.1.4
2047	5.912459	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.10? Tell 10.1.1.4
2048	5.912467	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.11? Tell 10.1.1.4
2049	5.912475	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.12? Tell 10.1.1.4
2050	5.912484	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.13? Tell 10.1.1.4
2051	5.912492	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.14? Tell 10.1.1.4
2052	5.912504	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.15? Tell 10.1.1.4
2053	6.012374	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.6? Tell 10.1.1.4
2054	6.012394	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.7? Tell 10.1.1.4
2055	6.012402	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.8? Tell 10.1.1.4
2056	6.012412	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.9? Tell 10.1.1.4
2057	6.012861	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.10? Tell 10.1.1.4
2058	6.014136	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.11? Tell 10.1.1.4
2059	6.014582	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.12? Tell 10.1.1.4
2060	6.014593	EdimaxTe_88:80:fd	Broadcast	ARP	who has 10.1.1.13? Tell 10.1.1.4

Tabla. 4.3. Verificación de Nmap

Otra de las exploraciones más comunes realizadas son las que buscan puertos UDP abiertos, a continuación se presenta un ejemplo de esta.

```
[root@localhost ~]# nmap -sT 10.1.1.0/24
```

Starting Nmap 5.00 (<http://nmap.org>) at 2009-10-15 17:10 ECT

A través de este comando se logra obtener una lista completa de los puertos UDP activos en un determinado host de la subred 10.1.1.0/24. Sin embargo para este caso no se despliega ningún resultado, veamos el tráfico capturado por wireshark en la interfaz del host 10.1.1.2 ante esta exploración.

No. -	Time	Source	Destination	Protocol	Info
18	0.309043	10.1.1.4	10.1.1.2	UDP	Source port: 41333 Destination port: 49182
19	0.309054	10.1.1.2	10.1.1.4	ICMP	Destination unreachable (Port unreachable)
20	0.309077	10.1.1.4	10.1.1.2	UDP	Source port: 41333 Destination port: 21710
21	0.309087	10.1.1.2	10.1.1.4	ICMP	Destination unreachable (Port unreachable)
22	0.309109	10.1.1.4	10.1.1.2	UDP	Source port: 41333 Destination port: 38037
23	0.309120	10.1.1.2	10.1.1.4	ICMP	Destination unreachable (Port unreachable)
24	0.309143	10.1.1.4	10.1.1.2	UDP	Source port: 41333 Destination port: 58178
25	0.309153	10.1.1.2	10.1.1.4	ICMP	Destination unreachable (Port unreachable)
26	0.313574	10.1.1.4	10.1.1.2	UDP	Source port: 41333 Destination port: 21702
27	0.313592	10.1.1.2	10.1.1.4	ICMP	Destination unreachable (Port unreachable)
28	0.313644	10.1.1.4	10.1.1.2	UDP	Source port: 41333 Destination port: 17359
29	0.313654	10.1.1.2	10.1.1.4	ICMP	Destination unreachable (Port unreachable)
30	0.313680	10.1.1.4	10.1.1.2	UDP	Source port: 41333 Destination port: 49360
31	0.313690	10.1.1.2	10.1.1.4	ICMP	Destination unreachable (Port unreachable)
32	0.314153	10.1.1.4	10.1.1.2	UDP	Source port: 41333 Destination port: 40847
33	0.314163	10.1.1.2	10.1.1.4	ICMP	Destination unreachable (Port unreachable)
34	0.314186	10.1.1.4	10.1.1.2	UDP	Source port: 41333 Destination port: 24644
35	0.314196	10.1.1.2	10.1.1.4	ICMP	Destination unreachable (Port unreachable)

Tabla. 4.4. Tráfico capturado por wireshark en la interfaz del host 10.1.1.2

La primera línea muestra al host 10.1.1.4 enviando un datagrama UDP sin ninguna información a la dirección IP 10.1.1.2. En este caso los puertos en este host están cerrados, y se envía un mensaje ICMP de puerto no alcanzable (*port unreachable*). Si el puerto está abierto, el host 10.1.1.4 no recibirá ninguna respuesta.

En el caso de detectar un elevado número de puertos UDP abiertos, el atacante podría concluir que existe un sistema cortafuegos entre su equipo y el objetivo. Para confirmar esta última posibilidad, se puede enviar un datagrama UDP al puerto cero. Esto tendría que generar una respuesta ICMP de puerto no alcanzable. No recibir esta respuesta significa que existe un dispositivo que filtra el tráfico.

2.1.3 Escuchas de red (Sniffers).

Constituye uno de los ataques iniciales contra las dos primeras capas del modelo TCP/IP. Se trata de un ataque realmente efectivo, puesto que permite la obtención de una gran cantidad de información sensible, a través de aplicaciones que se encargan de capturar e interpretar tramas y datagramas que circulan en la red.

Un *sniffer* es un programa que intercepta toda la información que pase por la interfaz de red a la que esté asociado. Una vez capturada, se podrá almacenar para su análisis posterior.

De esta forma, sin necesidad de acceso a ningún sistema de la red, un atacante podrá obtener información sobre cuentas de usuario, claves de acceso o incluso mensajes de correo electrónico en el que se envían contraseñas. Este tipo de técnica se conoce como *sniffing*.

3. TIPOS DE ATAQUES TCP/IP

3.1 Denegación de Servicio.

Un ataque de denegación de servicio (*Deny of service*) es un incidente en el cual un usuario o una organización es imposibilitada de acceder a un recurso o servicio, como por ejemplo, el acceso a una página web.

De forma más restrictiva, se pueden definir los ataques de denegación de servicio en redes IP como la consecución total o parcial (temporal o total) del cese de la prestación de servicio de un equipo conectado a la red.

Los ataques de denegación de servicio pueden ser provocados tanto por usuarios internos en el sistema como por usuarios externos. Dentro del primer grupo podríamos pensar en usuarios con pocos conocimientos que pueden colapsar el sistema o servicio inconscientemente.

En el segundo grupo se encuentran aquellos usuarios que han conseguido un acceso al sistema de forma ilegítima, falseando además la dirección de origen con el propósito de evitar la detección del origen real del ataque (mediante ataques de suplantación).

A continuación se realiza una exposición de algunos de los ataques de denegación de servicio más representativos.

3.1.1 Ataques de denegación de servicio distribuidos.

Se puede definir a los ataques de denegación de servicio distribuidos como un ataque DOS en el que existen múltiples equipos sincronizados de forma distribuida que se unen para atacar un mismo objetivo.

Los ataques de denegación de servicio distribuidos más representativos, prestando especial atención a su evolución histórica, al modelo distribuido de las fuentes que realizan el ataque, su sincronización y la forma en la que realizan la denegación de servicio se los ha distribuido en seis programas DDOS diferentes: Trinoo, Tribe Flood Network (TFN), TFN2K, Stacheldraht, Mstram y Shaft.

3.1.1.1 TRINOO.

TRINOO es un conjunto de herramientas *master-slave* utilizadas para sincronizar distintos equipos que cooperarán, de forma distribuida, en la realización de una denegación de servicio.

El primer paso para realizar un ataque con *TRINOO* consiste en la instalación de las herramientas en los equipos desde los que partirá el ataque. Para ello, el atacante necesitará obtener privilegios de administrador en estos equipos (que habrá conseguido mediante técnicas de *sniffing*)

Estos sistemas deberán ser equipos interconectados en grandes redes corporativas con un gran ancho de banda y en los que el origen del ataque pudiera pasar desapercibido entre cientos de sistemas dentro de la misma red. El atacante tratará de realizar un ataque de intrusión a un primer equipo de estas redes, desde donde continuará con la búsqueda de nuevos equipos vulnerables y procederá a su infección de igual manera como se realizó con el primer equipo.

Para realizar las intrusiones, el atacante llevará a cabo una búsqueda de vulnerabilidades en los equipos existentes en la red, generando una lista de equipos potencialmente débiles en los que tratará de introducirse y ejecutar las herramientas necesarias para provocar la escalada de privilegios.

Desde el primer equipo infectado por *TRINOO*, el atacante tratará de distribuir las herramientas a cada una de las demás máquinas infectadas. También se encargará de la ejecución de tareas periódicas para tratar de esconder los rastros de la intrusión que puedan delatar la entrada en el sistema y la detección del origen del ataque.

En la figura 3.2 podemos observar el diagrama de tres capas que conforma un ataque ejecutado mediante *TRINOO*. Se puede ver que a partir de un único

ordenador, el atacante podrá llegar a obtener toda una red de equipos a su disposición para la realización del ataque distribuido.

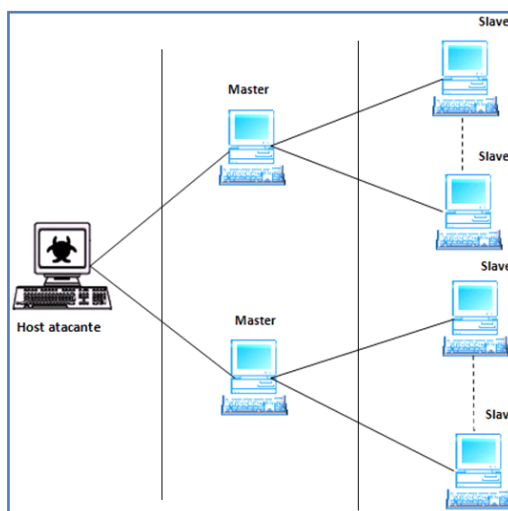


Figura. 4.2. Ataque ejecutado mediante *TRINOO*

La comunicación entre las distintas capas se realiza mediante conexiones TCP (fiables) para la parte *atacante-master*, y conexiones UDP (no fiables) para la parte *master-slave* y *slave-master*, en puertos específicos de cada máquina.

La comunicación siempre se inicia con la transmisión de una contraseña. Esto permite que ni el administrador del equipo ni el de otros atacantes puedan acceder al control de la red de ataques de *TRINOO*.

Los demonios de *TRINOO* situados en los equipos *master* y *slave* permiten la ejecución de comandos para iniciar, controlar y detener ataques de denegación tradicionales como *ICMP Flooding*, *SYN Flooding*, *UDP Flooding*, *Smurf*, etc. Para acceder a estos comandos, el atacante realizará una conexión Telnet en el puerto especificado en el siguiente esquema:

	Master	Slave
Atacante	27665/TCP	
Master		27444/UDP

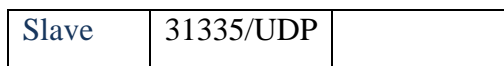


Figura. 4.3. Esquema de una conexión Telnet

3.1.1.2 TFN (Tribe Flood Network)

Es otra de las herramientas existentes para realizar ataques de Denegación de Servicio Distribuidos que utiliza un esquema *master-slave* para coordinar ataques de denegación tradicionales (*ICMP Flooding*, *SYN Flooding*, *UDP Flooding* y *Smurf*).

La comunicación entre el maestro y el demonio (slave) se lleva a cabo utilizando peticiones eco ICMP. Esta petición puede ordenar al demonio que realice una inundación UDP, TCP SYN, eco ICMP o un ataque Smurf. El maestro puede manipular el número de identificación IP y la carga de la respuesta eco ICMP para identificar el tipo de ataque a lanzar. TFN también puede falsear el origen IP para ocultar el origen del ataque.

El control de la red TFN se consigue mediante la ejecución directa de comandos utilizando conexiones *cliente-servidor* basadas en paquetes ICMP de tipo echo-reply.

La comunicación para el envío de comandos se realiza mediante un número binario de 16 bits en el campo de identificación de mensajes ICMP de tipo echo. El número de secuencia es una constante 0x0000 para enmascarar el mensaje ICMP como si fuera a una petición echo-request y pasar así desapercibido en el caso de existir en la red mecanismos de detección de ataques.

Este cambio en la comunicación, respecto a *TRIN00*, se debe a que muchos sistemas de monitorización para la protección de redes (dispositivos

cortafuegos, sistemas de detección de intrusos, . . .) pueden filtrar tráfico TCP y UDP que va hacia puertos determinados.

No obstante, la mayoría de sistemas dejan pasar mensajes ICMP de tipo echo utilizados para utilizar el comando *ping* y realizar así verificaciones de los equipos activos en la red.

Además, pocas herramientas de red muestran adecuadamente los mensajes ICMP, lo cual permite un camuflaje perfecto entre el tráfico normal de la red.

Otra diferencia respecto a *TRINOO* es que los clientes de TFN no están protegidos por contraseñas de acceso, con lo cual puede ser ejecutado sin restricciones por otros usuarios una vez instalado.

3.1.1.3 TFN2K

La arquitectura básica en la que existe un atacante que utiliza clientes para gobernar los distintos demonios instalados en las máquinas infectadas se mantiene, de forma que el control de este tipo de ataques mantiene la premisa de tener el máximo número de ordenadores segmentados. De esta forma, si un cliente es neutralizado, el resto de la red continúa bajo control.

Tribe Flood Network 2000 (TFN2k) añade una serie de características adicionales, de entre las que destacamos las siguientes:

- La comunicación *master-slave* se realizan ahora mediante protocolos TCP, UDP, ICMP o los tres a la vez de forma aleatoria.
- Los ataques continúan siendo los mismos (*ICMP Flooding*, *UDP Flooding*, *SYN Flooding* y *Smurf*). Aun así, el demonio se puede programar para que alterne entre estos cuatro tipos de ataque, para dificultar la detección por parte de sistemas de monitorización en la red.

- Las cabeceras de los paquetes de comunicación *master-slave* son ahora aleatorias, excepto en el caso de ICMP (donde siempre se utilizan mensajes de tipo echo-reply).
- De esta forma se evitaría una detección mediante patrones de comportamiento.
- Todos los comandos van cifrados. La clave se define en tiempo de compilación y se utiliza como contraseña para acceder al cliente.
- Cada demonio genera un proceso hijo por ataque, tratando de diferenciarse entre sí a través de los argumentos y parámetros que se pasan en el momento de ejecución.
- Además, se altera su nombre de proceso para hacerlo pasar por un proceso más del sistema.

3.1.1.4 Shaft

Otro conjunto de herramientas derivado de los dos anteriores (*TRIN00* y *TFN*) es *Shaft*.

La jerarquía utilizada por *Shaft* es similar a las demás herramientas analizadas. Una vez más, se basa en varios *masters* (denominados ahora *Shaftmasters*) que gobiernan a su vez diversos *slaves* (*Shaftnodes*).

Al igual que en los otros esquemas, el atacante se conecta mediante un programa cliente a los *Shaftmasters* desde donde inicia, controla y finaliza los ataques distribuidos.

Shaft utiliza mensajes UDP para transmitir información entre los *Shaftmasters* y los *Shaftnodes*.

Por otra parte, el atacante se conecta vía Telnet a un *Shaftmaster* utilizando una conexión fiable mediante TCP. Una vez conectado, utilizará una contraseña para autorizar su acceso.

	ShaftMaster	ShaftNode
Atacante	20432/TCP	
ShaftMaster		18753/UDP
ShaftNode	20433/UDP	

Figura. 4.4. Esquema de comunicaciones de Shaft

La comunicación entre *Shaftmasters* y *Shaftnodes* se realiza mediante UDP (que no es fiable). Por este motivo, Shaft utiliza *tickets* para poder mantener ordenada la comunicación y asignar a cada paquete una orden de secuencia.

3.2 Tipos de ataque DOS (Denegación de Servicio)

3.2.1 Ataque por fragmentación.

Un "**ataque por fragmentación**" consiste en saturar el tráfico de la red (*denegación de servicio*) para aprovechar el principio de fragmentación del protocolo IP.

La fragmentación IP puede plantear una serie de problemáticas relacionadas con la seguridad de la red, una ellas es la utilización de fragmentación malintencionada para burlar las técnicas básicas de inspección de datagramas IP.

En este caso, un atacante tratará de provocar intencionadamente una fragmentación en los datagramas que envía con el objetivo de que pasen desapercibidos por diferentes dispositivos de prevención y de detección de ataques que no tienen implementado el proceso de fragmentación y reensamblado de datagramas IP.

En el caso de los dispositivos de prevención más básicos (como, por ejemplo, routers con filtrado de paquetes), las decisiones para bloquear paquetes se basan generalmente en la información de cabecera de los paquetes (puertos TCP o UDP de destino, banderas de TCP, etc). Esto significa que los paquetes TCP y UDP fragmentados son susceptibles de burlar aquellos mecanismos de prevención que no implementen el proceso de reensamblado para poder tener una visión global del paquete que hay que bloquear.

Por otro lado, en el caso de dispositivos de prevención y detección más avanzados, las decisiones para detectar paquetes potencialmente peligrosos acostumbran a basarse nuevamente en la inspección de la cabecera del datagrama IP, así como en la parte de datos del paquete. Esto significa que la fragmentación se puede utilizar nuevamente para burlar este proceso de detección.

Con el objetivo de descubrir la MTU de la red e intentar así realizar fragmentación, el atacante puede utilizar el indicador de no fragmentación del datagrama IP. Cuando el indicador de no fragmentación está activado, el encaminador lo descubrirá, descartará el datagrama y devolverá el mensaje de error al equipo emisor. Este mensaje de error ICMP contiene la MTU de la red que requiere la fragmentación.

De esta forma, el atacante solo deberá construir datagramas con diferentes longitudes, con el indicador de fragmentación establecido, a la espera de recibir estos mensajes de error.

Para solucionar el uso de la fragmentación fraudulenta y garantizar una correcta inspección de paquetes, es necesaria la implementación del proceso de fragmentación y el reensamblado de datagramas en dispositivos de prevención y detección.

Esta solución puede suponer un coste adicional, ya que significa tener que examinar y almacenar cada fragmento. Aunque puede resultar muy costoso en cuanto a recursos (tiempo, proceso y memoria), será la única forma de asegurar que la inspección del paquete se ha realizado de forma correcta.

El ataque por fragmentación más conocido es *Teardrop*. Este método se basa en introducir información de compensación falsa en los paquetes fragmentados. En consecuencia, durante el reensamblado, quedan fragmentos vacíos o superpuestos que pueden desestabilizar el sistema.

3.2.2 IP Flooding.

El ataque de *IP Flooding* se basa en una inundación masiva de la red mediante datagramas IP.

Este ataque se realiza habitualmente en redes locales o en conexiones con un gran ancho de banda. Consiste en la generación de tráfico basura con el objetivo de conseguir la degradación del servicio. De esta forma, consume el ancho de banda disponible, reduciendo el desempeño en las comunicaciones existentes de toda la red.

Se puede pensar en la utilización de este ataque principalmente en redes locales cuyo control de acceso al medio es nulo y cualquier máquina puede enviar y recibir paquetes sin que se establezca ningún tipo de limitación en el ancho de banda que consume.

Los datagramas IP utilizados podrían corresponder a:

- **UDP.** Con el objetivo de generar peticiones sin conexión a ninguno de los puertos disponibles. Según la implementación de la pila TCP/IP de las máquinas involucradas, las peticiones masivas a puertos específicos UDP pueden llegar a causar el colapso del sistema.
- **ICMP.** Generando mensajes de error o de control de flujo.
- **TCP.** Para generar peticiones de conexión con el objetivo de saturar los recursos de red de la máquina atacada.

4. GENERACIÓN DE DIFERENTES TIPOS DE ATAQUES.

Para generar los diferentes tipos de ataques descritos a continuación, se utilizará la red de pruebas implementada la cual permitirá simular, analizar y determinar las características que se presentan ante un determinado ataque.

4.1 Spoofing

Se basa en la generación de paquetes IP con una dirección origen falsa. La idea de este ataque es muy sencilla: desde su equipo, el atacante simula la identidad de algún host de la red para conseguir acceso a un sistema objetivo que ha establecido algún tipo de confianza basada en el nombre o en la dirección IP del host suplantado o para burlar un dispositivo de filtrado que permite tráfico de paquetes con esa dirección IP.

En el Spoofing para que el atacante pueda conseguir su objetivo necesita establecer una comunicación falseada con la víctima, y evitar que el host suplantado interfiera en el ataque.

Este tipo de ataque o suplantación de identidad es muy común encontrarlo asociado con diferentes tipos de ataques mostrados más adelante, para ver su comportamiento se analizará primero un ataque relacionado con el establecimiento de una conexión TCP (TCP/SYN Flooding).

4.2 TCP/SYN Flooding

Algunos de los ataques que se utilizan en la actualidad se basan en no complementar intencionalmente el protocolo de intercambio TCP.

El ataque de **TCP/SYN Flooding** aprovecha del número de conexiones que está esperando el servidor para establecer un servicio en particular, denegando así el servicio.

Esto puede hacer que el sistema que es víctima del ataque sea incapaz de establecer cualquier conexión adicional para este servicio hasta que las conexiones que estén a la espera sean culminadas.

Hasta que se llegue a este límite, cada paquete SYN genera un SYN/ACK que permanecerá en la cola a la espera de establecerse. Es decir, cada conexión tiene un temporizador (un espacio de tiempo para el cual el sistema espera, el establecimiento de la conexión) que tiende a configurarse en un minuto.

Cuando se excede el límite de tiempo, se libera la memoria que mantiene el estado de esta conexión y la cuenta de la cola de servicios disminuye en una unidad. Después de alcanzar el límite, puede mantenerse completa la cola de servicios, evitando que el sistema establezca nuevas conexiones en este puerto con nuevos paquetes SYN.

Dado que el único propósito de la técnica es inundar la cola, no tiene ningún sentido utilizar la dirección IP real del atacante, ni tampoco devolver los SYN/ACK, puesto que de esta forma facilitaría que alguien pudiera llegar hasta el siguiendo la conexión. Por lo tanto, normalmente se falsea la dirección de origen del paquete (**spoofing**), modificando para ello la cabecera IP de los paquetes que intervendrán en el ataque de una inundación SYN.

4.3 Análisis del Tráfico Generado

Utilizando la red de pruebas implementada se realizará este ataque, el cual dependiendo de la ubicación del atacante tendrá algunas modificaciones, supongamos que el host 10.1.1.4 es el host del atacante el cual por razones evidentes no desea que se descubra el origen real del ataque, pues fácilmente siguiendo la conexión se puede llegar hacia él, entonces utilizando la técnica de **spoofing** suplantaré su verdadera identidad por el de algún host inactivo de la LAN, es importante que el host a utilizarse este fuera de servicio para que este no interfiera en el ataque.

Para comprender la inundación SYN se debe recordar algunas de las características de TCP que explota el ataque.

Para establecer una conexión TCP, las dos partes ejecutan el protocolo de intercambio tal y como se mostro en el capítulo 1, por ejemplo el Host A quiere establecer una sesión con B:

1. A envía un paquete SYN
2. B da acuse de recibo de SYN de A (SYN/ACK)
3. A da un acuse de recibo del SYN/ACK de B y se establece una conexión.

Cuando un atacante genera una inundación de paquetes SYN de TCP, no tiene ninguna intención de complementar el protocolo de intercambio, ni de establecer la conexión. Su objetivo es exceder los límites establecidos para el número de conexiones que están a la espera de establecerse para un servicio dado, veamos cómo se puede lograr esto: figura 4.5

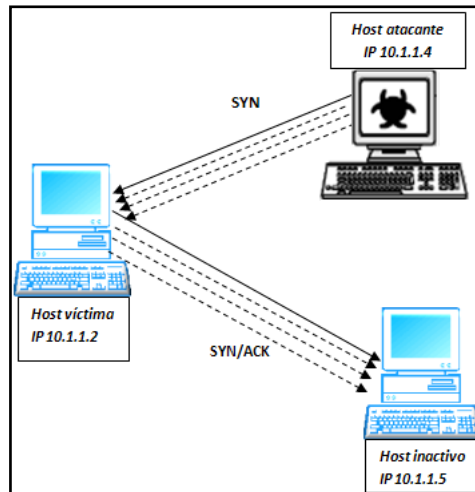


Figura. 4.5. Inundación de paquetes SYN de TCP

El atacante (IP 10.1.1.4) envía un paquete SYN indicando como dirección de origen la IP de un host inactivo en la red, así el host de destino (IP 10.1.1.2), responde a todas las peticiones de conexión con un SYN/ACK al host (IP 10.1.1.5) que simplemente lo ignorará por estar fuera de servicio (si no lo hiciera responderá con un RSH con lo cual la conexión se restearía y el ataque no sería posible).

A continuación se detalla el procedimiento utilizado para conseguir este ataque:

Mediante la herramienta nmap el atacante intentará ver que puerto se encuentran disponibles en el host víctima y así realizar este ataque cuyo efecto será denegar el servicio al cual el puerto está asociado.

Así por ejemplo ejecutando el siguiente comando desde el host atacante se tiene:

```
[root@localhost ~]# nmap -sS 10.1.1.2
```

```
Starting Nmap 5.00 ( http://nmap.org ) at 2009-11-22 16:14 ECT
```

```
Interesting ports on 10.1.1.2:
```

```
Not shown: 996 closed ports
```

```
PORT    STATE SERVICE
135/tcp  open  msrpc
3476/tcp open  unknown
3580/tcp open  unknown
61900/tcp open  unknown
MAC Address: 00:23:8B:7A:21:96 (Quanta Computer)
```

```
Nmap done: 1 IP address (1 host up) scanned in 14.65 seconds
```

Con lo cual se muestra que los puertos abiertos en el host 10.1.1.2 son: 135/tcp, 3476/tcp, 3580/tcp, 61900/tcp, sin embargo los puertos más explotados son los siguientes: 21, 23, 25, 69, 80, 110, 220 relacionados a los servicios y aplicaciones de mayor importancia y comúnmente usados en la actualidad.

Para poder efectuar este ataque se utilizara el puerto 135/tcp como ejemplo, el cual se lo visualiza así:

```
135/tcp  open  msrpc
```

Con lo cual se afirma que el puerto se encuentra abierto y disponible, entonces se satura al servidor (host victima) con excesivas peticiones SYN al puerto, para lo cual se utiliza la herramienta hping:

```
[root@localhost ~]# hping -a 10.1.1.5 -S -c 8000 --fast -p 135 10.1.1.2
HPING 10.1.1.2 (eth0 10.1.1.2): S set, 40 headers + 0 data bytes
```

```
--- 10.1.1.2 hping statistic ---
```

```
8000 packets tramitted, 0 packets received, 100% packet loss
```

El comando ejecutado permite suplantar la identidad de un host inactivo en la red (Spoofing), en este caso el host suplantado es el 10.1.1.5, y además envía 8000 peticiones SYN a la máxima velocidad posible dirigidas al puerto

135 con lo cual se consigue un exitoso ataque de denegación de servicio por inundación de paquetes SYN.

Para comprobar que el host víctima ha dejado de ofrecer el servicio relacionado al puerto atacado, el atacante una vez más se vale de nmap:

```
[root@localhost ~]# nmap -sS 10.1.1.2
```

```
Starting Nmap 5.00 ( http://nmap.org ) at 2009-11-22 21:05 ECT
```

```
Interesting ports on 10.1.1.2:
```

```
Not shown: 997 closed ports
```

```
PORT      STATE SERVICE
```

```
3476/tcp  open  unknown
```

```
3580/tcp  open  unknown
```

```
61900/tcp open  unknown
```

```
MAC Address: 00:23:8B:7A:21:96 (Quanta Computer)
```

```
Nmap done: 1 IP address (1 host up) scanned in 1.72 seconds
```

Cuyo resultado muestra que el puerto 135 ya no se encuentra disponible.

Un analista de red puede fácilmente detectar este tipo de ataque, si se captura en tráfico que entra por la interfaz del servidor se puede ver las siguientes características (tabla 3.5) que reflejan un ataque que utiliza las vulnerabilidades del estado de conexión del protocolo TCP.

No. -	Time	Source	Destination	Protocol	Info
540	31.658533	10.1.1.2	200.8.238.29	UDP	Source port: 3ds-lm Destination port: 14722
541	31.688619	10.1.1.5	10.1.1.2	TCP	conferencetalk > epmap [SYN] Seq=0 win=512 Len=0
542	31.789622	10.1.1.5	10.1.1.2	TCP	sesi-lm > epmap [SYN] Seq=0 win=512 Len=0
543	31.849989	24.36.69.182	10.1.1.2	UDP	Source port: 50731 Destination port: 3ds-lm
544	31.890546	10.1.1.5	10.1.1.2	TCP	houdini-lm > epmap [SYN] Seq=0 win=512 Len=0
545	31.991519	10.1.1.5	10.1.1.2	TCP	xmsg > epmap [SYN] Seq=0 win=512 Len=0
546	31.989182	10.1.1.2	74.206.204.42	TCP	8389 > 62914 [SYN] Seq=0 win=65535 Len=0 MSS=1460 WS=3 TSV=0
547	32.092474	10.1.1.5	10.1.1.2	TCP	fj-hdnet > epmap [SYN] Seq=0 win=512 Len=0
548	32.193445	10.1.1.5	10.1.1.2	TCP	h323gatedisc > epmap [SYN] Seq=0 win=512 Len=0
549	32.294448	10.1.1.5	10.1.1.2	TCP	h323gatestat > epmap [SYN] Seq=0 win=512 Len=0
550	32.395449	10.1.1.5	10.1.1.2	TCP	h323hostcall > epmap [SYN] Seq=0 win=512 Len=0
551	32.496413	10.1.1.5	10.1.1.2	TCP	caicci > epmap [SYN] Seq=0 win=512 Len=0
552	32.496437	QuantaCo_7a:21:96	Broadcast	ARP	who has 10.1.1.5? Tell 10.1.1.2
553	32.597326	10.1.1.5	10.1.1.2	TCP	hks-lm > epmap [SYN] Seq=0 win=512 Len=0
554	32.627444	200.8.238.29	10.1.1.2	UDP	Source port: 14722 Destination port: 3ds-lm
555	32.698331	10.1.1.5	10.1.1.2	TCP	pptp > epmap [SYN] Seq=0 win=512 Len=0
556	32.799293	10.1.1.5	10.1.1.2	TCP	csbphonemaster > epmap [SYN] Seq=0 win=512 Len=0
557	32.900315	10.1.1.5	10.1.1.2	TCP	iden-raip > epmap [SYN] Seq=0 win=512 Len=0

Tabla. 4.5. Tráfico que entra por la interfaz del servidor

Se observa que existe un excesivo número de peticiones SYN cuyo origen corresponde a una única dirección IP (10.1.1.5), que en principio se pensaría

que se trata del host atacante, pero en realidad como se vio anteriormente se trata de un ataque que utiliza spoofing para engañar al servidor, y permitir que este acepte dichas peticiones y lógicamente colapse en cuestión de segundos.

4.4 Smurf

El ataque Smurf aprovecha la capacidad de ICMP de enviar tráfico a la dirección de broadcast. Muchos hosts pueden escuchar y responder a peticiones eco ICMP enviadas a una dirección de broadcast. Esta capacidad es utilizada para ejecutar este ataque de denegación de servicio.

Primero el atacante debe crear una petición eco ICMP y dirigirla a una dirección de broadcast de una red intermedia con una IP de origen suplantada.

La dirección de origen corresponde a la dirección IP de la máquina que debe ser atacada. El campo de la dirección IP de destino corresponde a la dirección de broadcast, con lo cual cada host activo de esta red responde enviando una respuesta eco ICMP al host víctima, consumiendo todo el ancho de banda disponible y saturando el ordenador atacado.

Este tipo de ataque se puede dar si se cumplen las siguientes condiciones:

- El atacante envía muchas peticiones ICMP a la dirección de broadcast
- La red intermedia es grande y admite tráfico de broadcast interna
- El sitio objetivo tiene una conexión interna lenta.

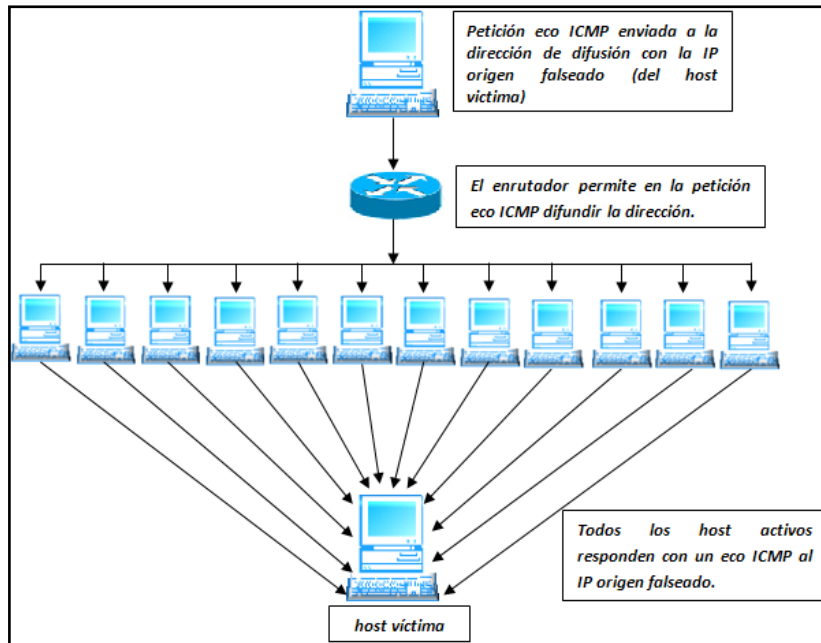


Figura. 4.6. Ataque Smurf

Para generar este tipo de ataque (figura 3.6) se utiliza la LAN interna 10.1.1.0/24 de la red de pruebas implementada, como medio de amplificación del ataque. El host víctima (IP 10.1.1.2), será inundado con respuestas ICMP de todos los host activos de la subred que respondan a peticiones ICMP falsas generadas desde el host atacante (IP 10.1.1.3).

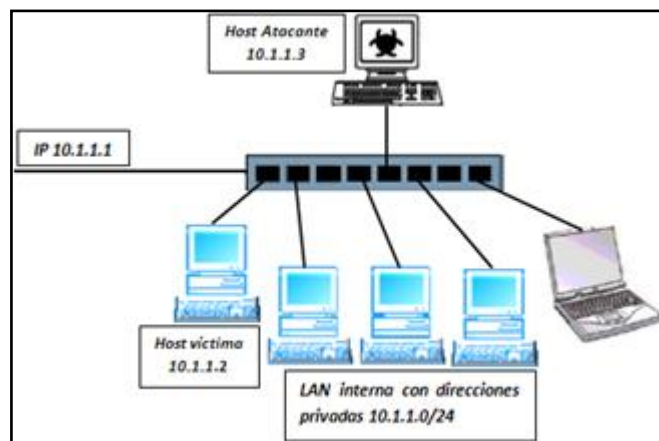


Figura. 4.7. Generación de Ataque Smurf

La herramienta utilizada para generar este tipo de ataque es hping2, con la cual se conseguirá suplantar (Spoofing) la dirección de la víctima para

generar las peticiones ICMP a toda la subred, utilizando la dirección de broadcast 10.1.1.255.

El host atacante utilizará el siguiente comando para lograr sus fines:

```
# hping -a 10.1.1.2 --icmp 10.1.1.255
```

```
HPING 10.1.1.255 (eth0 10.1.1.255): icmp mode set, 28 headers + 0 bytes
```

```
--- 10.1.1.255 hping statics ---
```

```
5648 packets tramitted, 0 packets received, 100 % packet loss
```

```
Round-trip min/avg/max = 0.0/0.0/0.0 ms
```

4.5 Análisis del Tráfico Generado

Se envía un paquete icmp (eco request) de longitud 28, con destino la dirección de broadcast 10.1.1.255, suplantando la dirección de origen (10.1.1.2).

Luego de finalizar la ejecución del comando se observa las estadísticas presentadas por hping en las cuales únicamente se muestran paquetes transmitidos, lo cual es lógico pues la dirección origen fue suplantada por la de la víctima la cual recibe todas las respuestas ICMP (reply) de los host activos de la subred.

Dado que hoy en día la mayoría de hosts de cualquier red TCP/IP por cuestiones de seguridad no responden a peticiones ICMP enviadas a la dirección de broadcast, hace que las probabilidades de sufrir un ataque smurf disminuyan.

En la red de pruebas los host de la LAN no responden a dichas peticiones ICMP, por lo cual sería imposible realizar o simular este tipo de ataque, sin embargo, se pueden presentar variantes en la utilización del comando hping, las cuales se muestran a continuación:

```
# hping -a 10.1.1.2 --icmp 10.1.1.4
HPING 10.1.1.4 (eth0 10.1.1.4): icmp mode set, 28 headers + 0 bytes
--- 10.1.1.4 hping statics ---
6385 packets tramitted, 0 packets received, 100 % packet loss
Round-trip min/avg/max = 0.0/0.0/0.0 ms
```

A través de este comando se logra el mismo efecto, con la diferencia que la dirección de destino es un solo host el cual responderá a la víctima con un paquete ICMP (reply), pero un solo host no causaría el daño esperado, por lo tanto se debe ejecutar el comando para cada host activo de la subred, incrementando la labor del atacante, pero logrando cumplir el objetivo.

```
# hping -a 10.1.1.2 -icmp 10.1.1.5
# hping -a 10.1.1.2 -icmp 10.1.1.6
# hping -a 10.1.1.2 -icmp 10.1.1.7
# hping -a 10.1.1.2 -icmp 10.1.1.8
.....
# hping -a 10.1.1.2 -icmp 10.1.1.254
```

A continuación con la ayuda de la herramienta de análisis y supervisión de redes wireshark se determinará las características y el comportamiento ante este tipo de ataque.

No. -	Time	Source	Destination	Protocol	Info
1	0.000000	10.1.1.6	10.1.1.2	ICMP	Echo (ping) reply
2	0.000460	10.1.1.10	10.1.1.2	ICMP	Echo (ping) reply
3	0.000468	10.1.1.8	10.1.1.2	ICMP	Echo (ping) reply
4	0.109952	10.1.1.9	10.1.1.2	ICMP	Echo (ping) reply
5	0.110906	10.1.1.7	10.1.1.2	ICMP	Echo (ping) reply
6	0.175911	10.1.1.11	10.1.1.2	ICMP	Echo (ping) reply
7	0.525889	10.1.1.250	10.1.1.2	ICMP	Echo (ping) reply
8	0.593837	10.1.1.254	10.1.1.2	ICMP	Echo (ping) reply
9	0.645791	10.1.1.12	10.1.1.2	ICMP	Echo (ping) reply
10	0.654769	10.1.1.13	10.1.1.2	ICMP	Echo (ping) reply
11	0.895805	10.1.1.5	10.1.1.2	ICMP	Echo (ping) reply
12	1.001784	10.1.1.6	10.1.1.2	ICMP	Echo (ping) reply
13	1.002243	10.1.1.10	10.1.1.2	ICMP	Echo (ping) reply
14	1.002248	10.1.1.8	10.1.1.2	ICMP	Echo (ping) reply
15	1.111680	10.1.1.9	10.1.1.2	ICMP	Echo (ping) reply
16	1.112654	10.1.1.7	10.1.1.2	ICMP	Echo (ping) reply
17	1.177654	10.1.1.11	10.1.1.2	ICMP	Echo (ping) reply
18	1.528639	10.1.1.250	10.1.1.2	ICMP	Echo (ping) reply

Tabla. 4.6. Tráfico que circula por la interfaz

La información capturada (tabla 3.6) corresponde al tráfico que circula por la interfaz de red del host victima (IP 10.1.1.2), se observa que todos los hosts activos de la subred utilizada para amplificar la transmisión de paquetes ICMP, responden simultáneamente con mensajes ICMP (reply) a la solicitud realizada por el host atacante, logrando así un efectivo ataque de denegación de servicio del tipo ICMP (smurf), cuyo efecto principal será el de consumir todo el ancho de banda disponible en la subred, colapsando al host victima y anulando gran cantidad de equipos involucrados en el ataque.

A continuación se observa la actividad de uno de los host involucrados en el ataque:

No. -	Time	Source	Destination	Protocol	Info
18	5.007609	10.1.1.2	10.1.1.5	ICMP	Echo (ping) request
19	5.007838	10.1.1.5	10.1.1.2	ICMP	Echo (ping) reply
20	6.009374	10.1.1.2	10.1.1.5	ICMP	Echo (ping) request
21	6.009573	10.1.1.5	10.1.1.2	ICMP	Echo (ping) reply
22	7.011092	10.1.1.2	10.1.1.5	ICMP	Echo (ping) request
23	7.011315	10.1.1.5	10.1.1.2	ICMP	Echo (ping) reply
24	8.012765	10.1.1.2	10.1.1.5	ICMP	Echo (ping) request
25	8.012997	10.1.1.5	10.1.1.2	ICMP	Echo (ping) reply
26	9.014534	10.1.1.2	10.1.1.5	ICMP	Echo (ping) request
27	9.014735	10.1.1.5	10.1.1.2	ICMP	Echo (ping) reply
28	10.015229	10.1.1.2	10.1.1.5	ICMP	Echo (ping) request
29	10.015457	10.1.1.5	10.1.1.2	ICMP	Echo (ping) reply
30	11.016963	10.1.1.2	10.1.1.5	ICMP	Echo (ping) request
31	11.017196	10.1.1.5	10.1.1.2	ICMP	Echo (ping) reply
32	12.018715	10.1.1.2	10.1.1.5	ICMP	Echo (ping) request
33	12.018911	10.1.1.5	10.1.1.2	ICMP	Echo (ping) reply
34	13.020406	10.1.1.2	10.1.1.5	ICMP	Echo (ping) request
35	13.020600	10.1.1.5	10.1.1.2	ICMP	Echo (ping) reply

Tabla 4.7: Host 10.1.1.5

Al parecer se trata de una simple petición ICMP (request) por parte del host 10.1.1.2, pero en realidad esta es realizada por el atacante el cual utiliza la dirección del host victima como origen, para engañar al host 10.1.1.5 y lograr que este responda al host víctima 10.1.1.2 con mensajes ICMP (reply), con el fin de inundar la red con grandes cantidades de mensajes ICMP y colapsar al host víctima.

La misma actividad mostrará cada uno de los hosts activos involucrados en el ataque. Mientras más grande (cantidad de host) sea la red mas eficiente y poderoso será el ataque smurf.

4.6 Teardrop

El protocolo IP especifica unos campos en su cabecera encargados de señalar si el datagrama IP está fragmentado y la posición que ocupa dentro del datagrama original.

El ataque **Teardrop** intentará realizar una utilización fraudulenta de la fragmentación IP para poder confundir al sistema operativo en la reconstrucción del datagrama original y colapsar así el sistema.

Supongamos que deseamos enviar un fichero de 1024 bytes a una red con un MTU (*Maxim Transfer Unit*) de 512 bytes. Será suficiente enviar dos fragmentos de 512 bytes:

El objetivo de *Teardrop* será realizar las modificaciones necesarias en los campos de posición y longitud para introducir incoherencias cuando se produzca la reconstrucción del datagrama original:

De esta forma, *Teardrop* y sus variantes directas conseguirán que el datagrama no sea reconstruido y provoque, si se realiza de manera distribuida que el sistema colapse.

Otra posibilidad consiste en enviar centenares de fragmentos modificados malintencionadamente, con el objetivo de saturar la pila de protocolo IP del equipo atacado (a causa de una superposición de distintos datagramas IP).

4.7 Ping of death

El ataque de denegación de servicio "*ping de la muerte*" (*ping of death*) es uno de los ataques más conocidos. Al igual que otros ataques de denegación existentes, utiliza una definición de longitud máxima de datagrama.

El **ataque ping de la muerte** es uno de los ataques de red más antiguos. El principio de este ataque consiste simplemente en crear un datagrama IP cuyo tamaño total supere el máximo autorizado (65.536 bytes). Cuando un paquete con estas características se envía a un sistema que contiene una pila vulnerable de protocolos TCP/IP, éste produce la caída del sistema.

Los sistemas más modernos ya no son vulnerables a este tipo de ataque.

4.8 Análisis del Tráfico Generado

La longitud máxima de un datagrama IP es de 65535 bytes, incluyendo la cabecera del paquete (20 bytes) y partiendo de la base de que no hay opciones especiales especificadas. Por otra parte, el protocolo ICMP tiene una cabecera de 8 bytes. De esta forma, si se desea construir un mensaje ICMP tenemos disponibles $65535 - 20 - 8 = 65507$ bytes.

Debido a la posibilidad de fragmentación de IP, si es necesario enviar más de 65535 bytes, el datagrama IP se fragmentará y se reensamblará en el destino.

El ataque ping de la muerte se basa en la posibilidad de construir, mediante el comando ping, un datagrama IP superior a los 65535 bytes, fragmentado en N trozos, con el objetivo de provocar incoherencias en el proceso de reensamblado.

Si, por ejemplo, se construye un mensaje ICMP de tipo echo-request de 65510 bytes mediante el comando `ping -s 65510`, los datos ICMP podrán ser enviados en un único paquete fragmentado en N trozos (según la MTU de la red), pero pertenecientes al mismo datagrama IP. Si se realiza la suma de los distintos campos del datagrama, se verá que los 20 bytes de cabecera IP más los 8 bytes de cabecera ICMP, junto con los datos ICMP (65510 bytes) ocuparán 65538 bytes. De esta forma, el ataque consigue provocar un

desbordamiento de 3 bytes. Este hecho provocará que al reconstruir el paquete original en el destino, se producirán errores, causando la degradación total del sistema atacado.

A continuación se observa que es posible enviar un paquete eco ilegal con más de 65.500 octetos de datos debido a la forma en que se realiza la fragmentación. La fragmentación confía en un valor de desplazamiento de cada fragmento para determinar dónde se ubica cada fragmento tras la reconstrucción. Por tanto en el último fragmento es posible combinar un desplazamiento válido con un tamaño de fragmento de forma que la suma (desplazamiento + tamaño) > 65.535. Como los equipos normales no procesan el paquete hasta que tienen todos los fragmentos y tratan de reconstruirlos, hay posibilidades de que se produzca una sobrecarga de las variables internas de 16 bits, lo que nos puede llevar a caídas de sistemas, reinicios, volcados de núcleo y cosas así.

La siguiente simulación muestra la forma en la que se puede generar este tipo de ataque:

```
[root@localhost ~]# ping -s 65510 10.1.1.2
PING 10.1.1.2 (10.1.1.2) 65510(65528) bytes of data.
65508 bytes from 10.1.1.2: icmp_seq=2 ttl=48 time=11.8 ms
65508 bytes from 10.1.1.2: icmp_seq=3 ttl=48 time=11.9 ms
65508 bytes from 10.1.1.2: icmp_seq=4 ttl=48 time=11.9 ms
65508 bytes from 10.1.1.2: icmp_seq=5 ttl=48 time=11.8 ms
65508 bytes from 10.1.1.2: icmp_seq=6 ttl=48 time=11.9 ms
65508 bytes from 10.1.1.2: icmp_seq=7 ttl=48 time=11.9 ms
65508 bytes from 10.1.1.2: icmp_seq=8 ttl=48 time=11.9 ms
65508 bytes from 10.1.1.2: icmp_seq=9 ttl=48 time=11.9 ms
65508 bytes from 10.1.1.2: icmp_seq=10 ttl=48 time=11.8 ms
65508 bytes from 10.1.1.2: icmp_seq=11 ttl=48 time=11.8 ms
65508 bytes from 10.1.1.2: icmp_seq=306 ttl=48 time=16.8 ms
65508 bytes from 10.1.1.2: icmp_seq=307 ttl=48 time=16.7 ms
65508 bytes from 10.1.1.2: icmp_seq=308 ttl=48 time=16.8 ms
```

65508 bytes from 10.1.1.2: icmp_seq=309 ttl=48 time=16.8 ms
65508 bytes from 10.1.1.2: icmp_seq=310 ttl=48 time=16.8 ms
65508 bytes from 10.1.1.2: icmp_seq=311 ttl=48 time=16.9 ms
65508 bytes from 10.1.1.2: icmp_seq=313 ttl=48 time=16.9 ms
65508 bytes from 10.1.1.2: icmp_seq=314 ttl=48 time=16.8 ms
65508 bytes from 10.1.1.2: icmp_seq=372 ttl=48 time=11.9 ms
65508 bytes from 10.1.1.2: icmp_seq=373 ttl=48 time=11.9 ms

--- 10.1.1.2 ping statistics ---

373 packets transmitted, 372 received, 0% packet loss, time 372201ms
rtt min/avg/max/mdev = 11.837/13.185/16.938/2.051 ms

CAPITULO 5

ANÁLISIS ESTADÍSTICO DE LOS DATOS DEL TRÁFICO TCP/IP DE LA RED DEL DEPARTAMENTO DE ELECTRICA Y ELECTRÓNICA DE LA ESPE (DEEE)

1. INTRODUCCIÓN

En este capítulo básicamente se analiza información del tráfico que circula por la red del departamento de eléctrica y electrónica de la ESPE, el cual constituye un primer análisis de este tipo, enfocándose en la importancia de conocer el comportamiento de la carga de tráfico de una red para la toma de decisiones ante una eventual anomalía. El estudio se hizo con información y arquitectura de la red del DEEE durante la semana del miércoles 30 de septiembre al viernes 9 de octubre de 2009, se hace esta aclaración por cuanto el diseño y la estructura de la red pueden ser modificados en cualquier momento por administradores de red.

Se utilizan métodos estadísticos que permitan construir características y perfiles que determinen el modelo del comportamiento normal de operación de la red, creando así una herramienta de seguridad que permita detectar posibles intrusiones.

2. CONCEPTOS ESTADÍSTICOS

2.1 Selección de la muestra

Una de las principales funciones de la estadística consiste en la descripción de los datos; ya sea por medio de medidas (estimadores), gráficos o tablas en las que se puedan apreciar claramente el comportamiento y las tendencias de la información recopilada.

Debemos recordar que la estadística es un sistema o método empleado en la recolección, organización, análisis e interpretación de los datos. Esta ciencia se divide en dos fases; la primera corresponde a la *Estadística descriptiva*, cuya finalidad es agrupar y representar la información de forma ordenada, de tal manera que nos permita identificar rápidamente aspectos característicos del comportamiento de los datos. La segunda fase corresponde a la *Estadística de Inferencia*, la cual busca dar explicación al comportamiento o hallar conclusiones de un amplio grupo de eventos, objetos o sucesos a través del análisis de una pequeña fracción de sus componentes (*Muestra*).

En este capítulo se centra exclusivamente en la aplicación de la *Estadística Descriptiva* y los procedimientos que la componen, como las medidas de tendencia central, medidas de distribución y las medidas de dispersión. Antes de conocer cada una de estas medidas es necesario resaltar la diferencia entre *Población* y *Muestra*.

2.1.1 Población.

Se denomina *Población* al total de los elementos que componen un conjunto, el cual es el objeto de interés de un estudio.

Las poblaciones pueden ser finitas o infinitas de acuerdo si se conoce el total de los elementos que la componen o no. Generalmente es bastante difícil realizar un estudio con el total de la población, ya sea por que es demasiado grande, requiere demasiado tiempo para su análisis, los costos son muy elevados, se desconoce el total de elementos, etc.

Por estas razones se suele sustraer una pequeña fracción de la población para realizar los análisis; de tal manera que las conclusiones que se extraigan sobre la fracción sean aplicables a la población. A esta fracción se le denomina **Muestra** y cada uno de los procedimientos estadísticos presenta algunas variaciones en sus ecuaciones de acuerdo si los datos representan muestras o poblaciones.

2.1.2 Muestreo.

Esto no es más que la técnica o procedimiento empleado para obtener una o más muestras de una población.

Este se realiza una vez que se ha establecido un marco muestral representativo de la población.

Al tomar varias muestras de una población, las estadísticas que calculamos para cada muestra no necesariamente serían iguales, y lo más probable es que variaran de una muestra a otra.

El muestreo nos da la posibilidad de conocer lo que queremos saber de la población porque al analizar un número proporcional de ésta, seremos capaces de conocer sus opiniones acerca de las cuestiones que teníamos, pues los resultados arrojados por la muestra son una proyección de la opinión general.

2.2 Técnicas del muestreo

2.2.1 Muestreo probabilístico.

Se caracteriza por proporcionar la misma probabilidad a toda la población de formar parte de la muestra. Puede ser: aleatorio simple, sistemático, estratificado, por agrupamientos y algunas otras técnicas.

2.2.2 Muestreo no probabilístico.

Esta técnica se basa en el juicio personal del investigador para la selección de los elementos, dividiéndose en las siguientes clasificaciones: por conveniencia, por juicio, por cuota o por bola de nieve.

2.2.3 Tamaño de la muestra.

El cálculo del tamaño de la muestra es uno de los aspectos a concretar en esta fase del análisis y determina el grado de credibilidad que concederemos a los resultados obtenidos.

Una fórmula muy extendida que orienta sobre el cálculo del tamaño de la muestra para datos globales es la siguiente:

$$n = \frac{PQZ^2N}{E^2(N-1) + Z^2PQ}$$

- n=tamaño de la muestra
- P=Probabilidad de éxito
- Q= 1 – P (probabilidad de fracaso)
- N=Total de la población

- Z^2 =es una constante que depende del nivel de confianza que asignemos. El nivel de confianza indica la probabilidad de que los resultados de nuestra investigación sean ciertos.

Los valores k más utilizados y sus niveles de confianza son:

Z	1.15	1.28	1.44	1.65	1.96	2	2.58
Nivel de confianza	75%	80%	85%	90%	95%	95.5%	99%

- e^2 =es el error muestral deseado. El error muestral es la diferencia que puede haber entre el resultado que obtenemos preguntando a una muestra de la población y el que obtendríamos si preguntáramos al total de ella.

3. ESTADÍSTICA DESCRIPTIVA

3.1 Utilización de herramientas de office 2007

Para el análisis estadístico se utilizó la herramienta Excel 2007 de Microsoft Office, se realizó la selección aleatoria de la muestra, se agruparon los datos a través de la utilización de la tabla dinámica que permitió definir con exactitud la cantidad de datos y tipo de protocolo que circularon por la red.

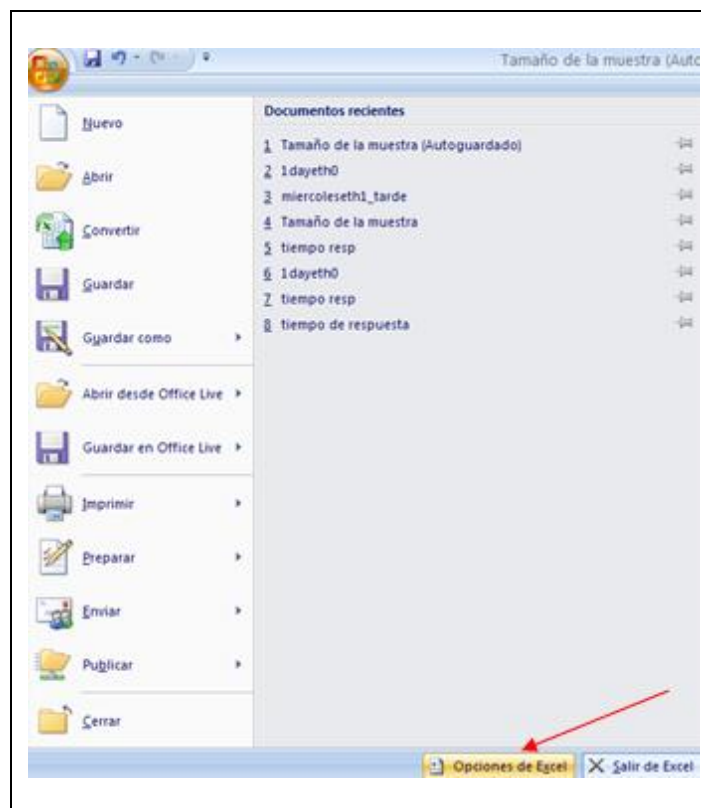
Así mismo graficar con exactitud la tendencia de cada evento, el análisis descriptivo y el análisis de regresión.

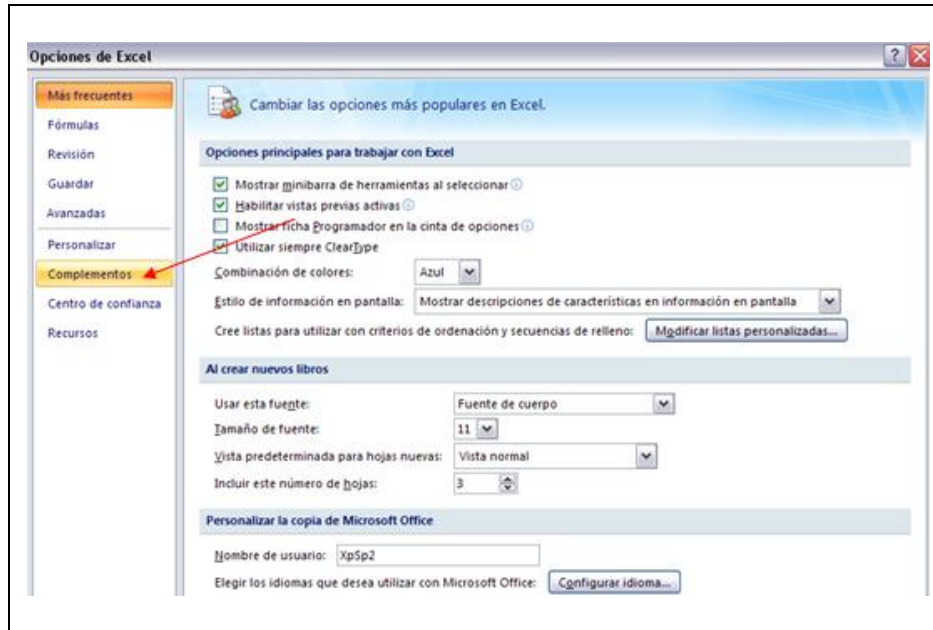
Para habilitar el análisis estadístico en la base de datos Excel 2007, debe activarse la opción Herramientas para Análisis de Datos tal como se ilustra a continuación:

3.1.1 Herramientas / Análisis de datos.

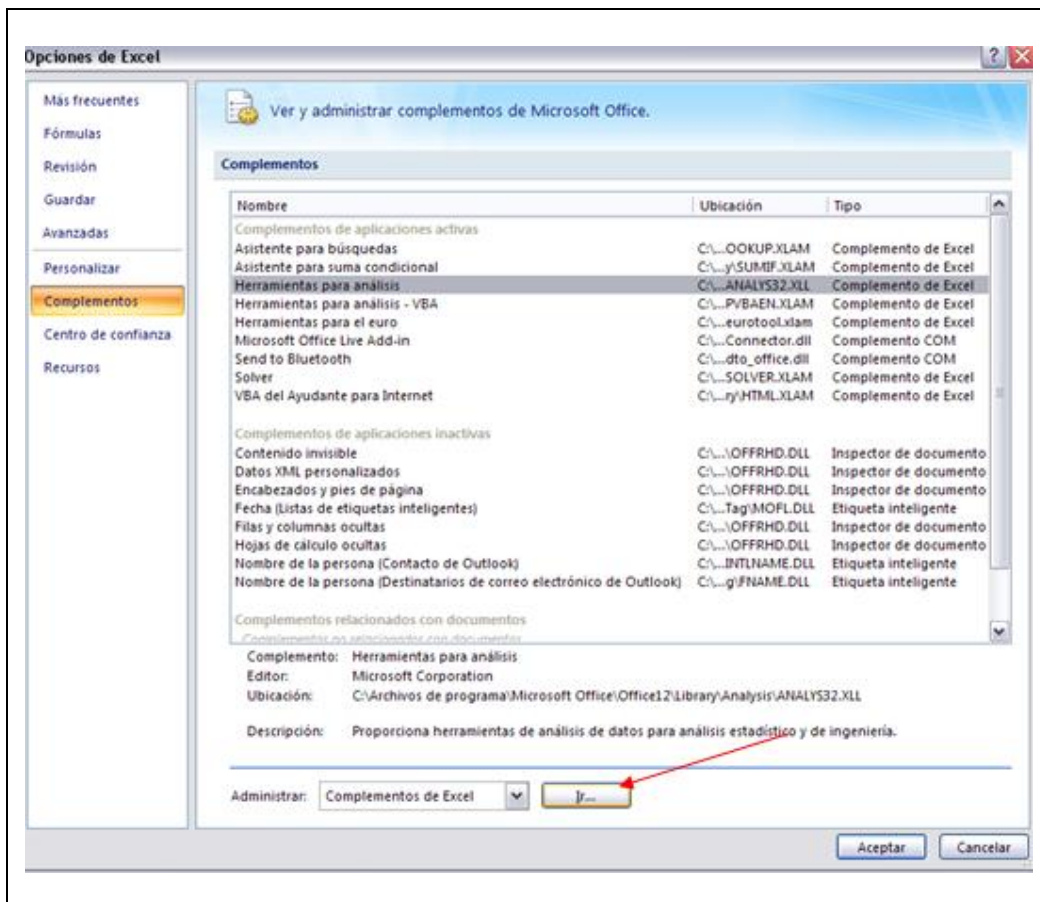
Si no se tiene activada la herramienta de **análisis de datos**, se debe activarla. Para ello se debe seguir los siguientes pasos:

1. Clic **Archivo**, luego seleccionar Opciones de Excel, **Complementos del Backstage de Excel 2007**.

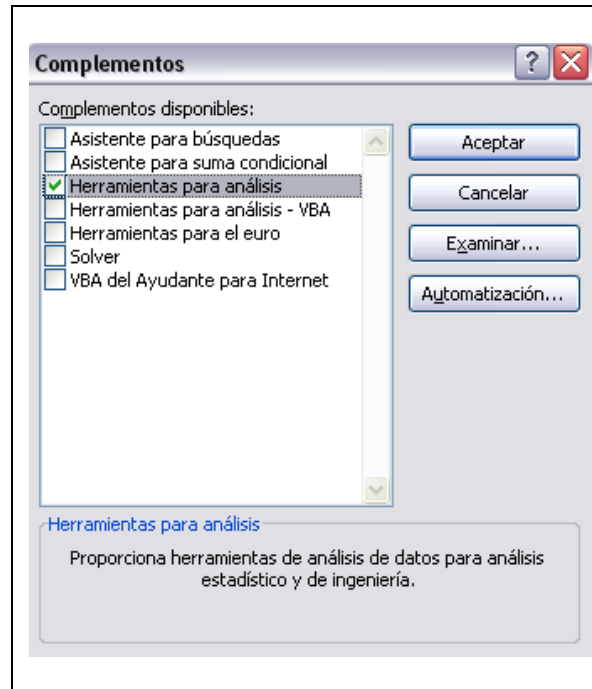




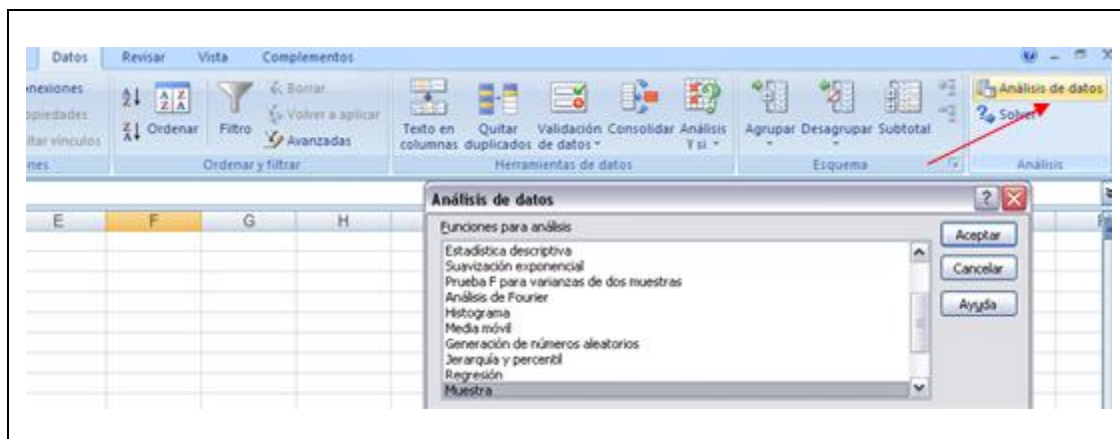
2. En el cuadro que se despliega se busca y marca el cuadrado correspondiente a **Herramientas para análisis**, y se da clic en Ir (Ubicado en la parte central inferior).



- Finalmente se selecciona la opción **Herramienta para análisis**, y se da clic en Aceptar.



- Navegar a la pestaña **Datos** y observar que se ha agregado el grupo **Análisis** y dentro tiene el botón para acceder a **Análisis de datos**.



3.2 Estadígrafos de posición

3.2.1 Medidas tendencia central.

Este tipo de medidas nos permiten identificar y ubicar el punto (valor) alrededor del cual se tienden a reunir los datos (“Punto central”). Estas medidas aplicadas a las características de las unidades de una muestra se les denomina estimadores o estadígrafos; mientras que aplicadas a poblaciones se les denomina parámetros o valores estadísticos de la población. Los principales métodos utilizados para ubicar el punto central son la media, la mediana y la moda.

3.2.1.1 Media

Es la medida de posición central más utilizada, la más conocida y la más sencilla de calcular, debido principalmente a que sus ecuaciones se prestan para el manejo algebraico, lo cual la hace de gran utilidad. Su principal desventaja radica en su sensibilidad al cambio de uno de sus valores o a los valores extremos demasiado grandes o pequeños. La media se define como la suma de todos los valores observados, dividido por el número total de observaciones.

$$\text{Media Aritmetica} = \frac{\text{Suma de todos los valores observados}}{\text{Número total de observaciones}}$$

Ecuación 4-1

Cuando los valores representan una población la ecuación se define como:

$$\bar{u} = \frac{x_1 + x_2 + x_3 + \dots + x_n}{N} = \frac{\sum_{i=1}^n x_i}{N}$$

Donde (\bar{u}) representa la media, (N) representa el tamaño de la población y (x_i) representa cada uno de los valores de la población. Ya que en la mayoría de los casos se trabajan con muestras de la población.

Ecuación 4-2

Todas las ecuaciones que se presenten a continuación serán representativas para las muestras. La media aritmética para una muestra está determinada como:

$$\bar{X} = \frac{x_1 + x_2 + x_3 + \dots + x_n}{n} = \frac{\sum_{i=1}^n x_i}{n}$$

Donde (\bar{x}) representa la Media para la muestra, (n) el tamaño de la muestra y (x_i) representa cada uno de los valores observados. Esta fórmula únicamente es aplicable si los datos se encuentran desagrupados; en caso contrario debemos calcular la media mediante la multiplicación de los diferentes valores por la frecuencia con que se encuentren dentro de la información; es decir:

Ecuación 4-3

$$\bar{X} = \frac{\sum_{i=1}^n y_i n_i}{n}$$

Donde (y_i) representa el punto medio de cada observación, (n_i) es la frecuencia o número de observaciones en cada clase y (n) es el tamaño de la muestra siendo igual a la suma de las frecuencias de cada clase.

Es importante resaltar que existe una gran variedad de medias como la *Media geométrica*, la *Media ponderada*, la *Media cuadrática*, etc. Por el momento sólo hacemos énfasis en la media aritmética ya que es la más utilizada, aunque se recomienda a los lectores profundizar en estos temas.

3.2.1.2 Mediana.

Con esta medida podemos identificar el valor que se encuentra en el centro de los datos, es decir, nos permite conocer el valor que se encuentra exactamente en la mitad del conjunto de datos, después que las observaciones se han ubicado en serie ordenada. Esta medida nos indica que la mitad de los datos se encuentran por debajo de este valor y la otra mitad por encima del mismo. Para determinar la posición de la mediana se utiliza la fórmula:

Ecuación 4-4

$$\textit{Posición de la mediana} = \frac{n + 1}{2}$$

En conclusión la mediana nos indica el valor que separa los datos en dos fracciones iguales con el cincuenta por ciento de los datos cada una. Para las muestras que cuentan con un número impar de observaciones o datos, la mediana dará como resultado una de las posiciones de la serie ordenada; mientras que para las muestras con un número par de observaciones se debe promediar los valores de las dos posiciones centrales.

3.2.1.3 Moda.

La medida modal nos indica el valor que más veces se repite dentro de los datos; es decir, si tenemos la serie ordenada (2, 2, 4 y 7), el valor que más veces se repite es el número 2 quien sería la moda de los datos. Es posible que en algunas ocasiones se presente dos valores con la mayor frecuencia, lo

cual se denomina *Bimodal* o en otros casos más de dos valores, lo que se conoce como *multimodal*.

3.2.1.4 Resumen.

Las **Medidas de tendencia central**, nos permiten identificar los valores más representativos de los datos, de acuerdo a la manera como se tienden a concentrar. La **Media** nos indica el promedio de los datos; es decir, nos informa el valor que obtendría cada uno de los eventos si se distribuyeran los valores en partes iguales. La **Mediana** por el contrario nos informa el valor que separa los datos en dos partes iguales, cada una de las cuales cuenta con el cincuenta por ciento de los datos. Por último la **Moda** nos indica el valor que más se repite dentro de los datos.

3.2.2 Medidas de Dispersión.

Así como las medidas de tendencia central nos permiten identificar el punto central de los datos, las *Medidas de dispersión* nos permiten reconocer que tanto se dispersan los datos alrededor del punto central; es decir, nos indican cuanto se desvían las observaciones alrededor de su promedio aritmético (Media). Este tipo de medidas son parámetros informativos que nos permiten conocer como los valores de los datos se reparten a través de eje X, mediante un valor numérico que representa el promedio de dispersión de los datos. Las medidas de dispersión más importantes y las más utilizadas son la *Varianza* y la *Desviación estándar* (o Típica).

3.2.2.1 Varianza

Esta medida nos permite identificar la diferencia promedio que hay entre cada uno de los valores respecto a su punto central (*Media \bar{x}*). Este promedio es calculado, elevando cada una de las diferencias al cuadrado (Con el fin de

eliminar los signos negativos), y calculando su promedio o media; es decir, sumado todos los cuadrados de las diferencias de cada valor respecto a la media y dividiendo este resultado por el número de observaciones que se tengan. Si la varianza es calculada a una población (Total de componentes de un conjunto), la ecuación sería:

Ecuación 4-5

$$\sigma^2 = \frac{(x_1 - \bar{u})^2 + (x_2 - \bar{u})^2 + (x_3 - \bar{u})^2 + \dots + (x_n - \bar{u})^2}{N} = \frac{\sum(x_i - \bar{u})^2}{N}$$

Donde (σ^2) representa la varianza, (x_i) representa cada uno de los valores, (\bar{u}) representa la media poblacional y (N) es el número de observaciones ó tamaño de la población. En el caso que estemos trabajando con una muestra la ecuación que se debe emplear es:

Ecuación 4-6

$$S^2 = \frac{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + (x_3 - \bar{x})^2 + \dots + (x_n - \bar{x})^2}{(n - 1)} = \frac{\sum(x_i - \bar{x})^2}{(n - 1)}$$

Donde (S^2) representa la varianza, (x_i) representa cada uno de los valores, (\bar{x}) representa la media de la muestra y (n) es el número de observaciones ó tamaño de la muestra. Si nos fijamos en la ecuación, notaremos que se le resta uno al tamaño de la muestra; esto se hace con el objetivo de aplicar una pequeña medida de corrección a la varianza, intentando hacerla más representativa para la población. Es necesario resaltar que la varianza nos da como resultado el promedio de la desviación, pero este valor se encuentra elevado al cuadrado.

3.2.2.2 Desviación estándar o Típica

Esta medida nos permite determinar el promedio aritmético de fluctuación de los datos respecto a su punto central o media. La desviación estándar nos da como resultado un valor numérico que representa el promedio de diferencia que hay entre los datos y la media. Para calcular la desviación estándar basta con hallar la raíz cuadrada de la varianza, por lo tanto su ecuación sería:

Ecuación 4-7

$$S = \sqrt{S^2}$$

3.2.3 Medidas de Distribución.

Las medidas de distribución nos permiten identificar la forma en que se separan o aglomeran los valores de acuerdo a su representación gráfica. Estas medidas describen la manera como los datos tienden a reunirse de acuerdo con la frecuencia con que se hallen dentro de la información. Su utilidad radica en la posibilidad de identificar las características de la distribución sin necesidad de generar el gráfico. Sus principales medidas son la *Asimetría* y la *Curtosis*.

3.2.3.1 Asimetría.

Esta medida nos permite identificar si los datos se distribuyen de forma uniforme alrededor del punto central (Media aritmética). La asimetría presenta tres estados diferentes (Fig.5.1), cada uno de los cuales define de forma concisa como están distribuidos los datos respecto al eje de asimetría. Se

dice que la *asimetría es positiva* cuando la mayoría de los datos se encuentran por encima del valor de la media aritmética, la curva es *Simétrica* cuando se distribuyen aproximadamente la misma cantidad de valores en ambos lados de la media y se conoce como *asimetría negativa* cuando la mayor cantidad de datos se aglomeran en los valores menores que la media.

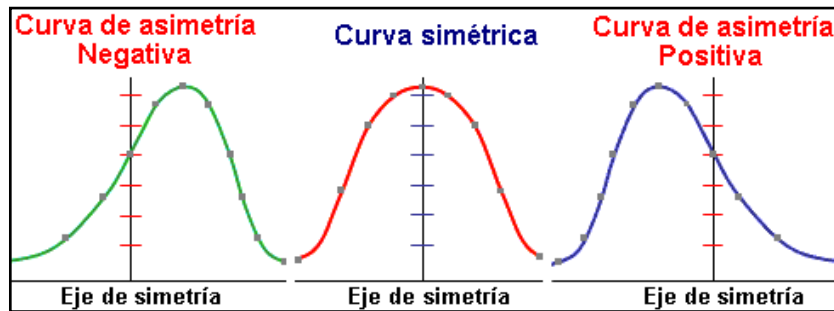


Figura. 5.1. Estados de Asimetría

El *Coefficiente de asimetría*, se representa mediante la ecuación matemática:

Ecuación 4-8

$$g_1 = \frac{\frac{1}{n} \sum (x_i - \bar{x})^3 * n_i}{\left(\frac{1}{n} \sum (x_i - \bar{x})^2 * n_i \right)^{\frac{3}{2}}}$$

Donde (g_1) representa el coeficiente de asimetría de Fisher, (x_i) cada uno de los valores, (\bar{x}) la media de la muestra y (n_i) la frecuencia de cada valor. Los resultados de esta ecuación se interpretan:

- ($g_1 = 0$): Se acepta que la distribución es Simétrica, es decir, existe aproximadamente la misma cantidad de valores a los dos lados de la media. Este valor es difícil de conseguir por lo que se tiende a tomar los valores que son cercanos ya sean positivos o negativos (± 0.4).

- ($g_1 > 0$): La curva es asimétricamente positiva por lo que los valores se tienden a reunir más en la parte izquierda que en la derecha de la media.
- ($g_1 < 0$): La curva es asimétricamente negativa por lo que los valores se tienden a reunir más en la parte derecha de la media.

Desde luego entre mayor sea el número (Positivo o Negativo), mayor será la distancia que separa la aglomeración de los valores con respecto a la media.

3.2.3.2 Curtosis.

Esta medida determina el grado de concentración que presentan los valores en la región central de la distribución. Por medio del *Coefficiente de Curtosis*, podemos identificar si existe una gran concentración de valores (*Leptocúrtica*), una concentración normal (*Mesocúrtica*) ó una baja concentración (*Platicúrtica*).

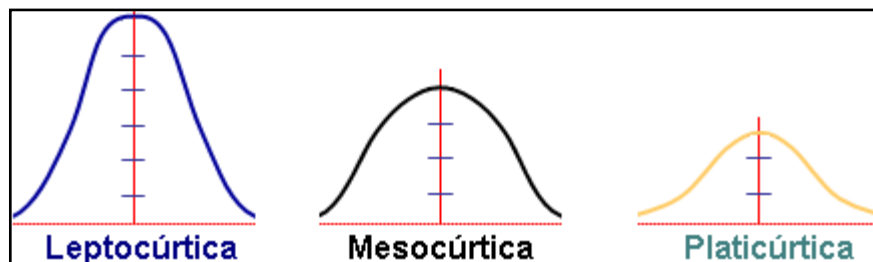


Figura. 5.2. Concentración de Valores

Para calcular el coeficiente de Curtosis se utiliza la ecuación:

$$g_2 = \frac{\frac{1}{n} \sum (x_i - \bar{x})^4 * n_i}{\left(\frac{1}{n} \sum (x_i - \bar{x})^2 * n_i \right)^2} - 3$$

Donde (g_2) representa el coeficiente de Curtosis, (x_i) cada uno de los valores, (\bar{x}) la media de la muestra y (n_i) la frecuencia de cada valor. Los resultados de esta fórmula se interpretan:

- $(g_2 = 0)$ la distribución es *Mesocúrtica*: Al igual que en la asimetría es bastante difícil encontrar un coeficiente de Curtosis de cero (0), por lo que se suelen aceptar los valores cercanos (± 0.4 aprox.).
- $(g_2 > 0)$ la distribución es *Leptocúrtica*
- $(g_2 < 0)$ la distribución es *Platicúrtica*

Cuando la distribución de los datos cuenta con un coeficiente de asimetría $(g_1 = \pm 0.4)$ y un coeficiente de Curtosis de $(g_2 = \pm 0.4)$, se le denomina Curva Normal. Este criterio es de suma importancia ya que para la mayoría de los procedimientos de la estadística de inferencia se requiere que los datos se distribuyan normalmente.

La principal ventaja de la distribución normal radica en el supuesto que el 94% de los valores se encuentra dentro de una distancia de dos desviaciones estándar de la media aritmética (Fig.5.3); es decir, si tomamos la media y le sumamos dos veces la desviación y después le restamos a la media dos desviaciones, el 94% de los casos se encontraría dentro del rango que compongan estos valores.

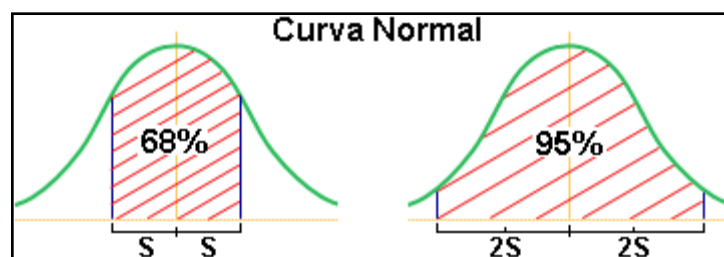


Figura. 5.3. Distancia de dos desviaciones estándar de la media aritmética

Desde luego, los conceptos vistos hasta aquí, son sólo una pequeña introducción a las principales medidas de Estadística Descriptiva.

3.3 Correlación y Regresión Lineal

El objetivo de estas técnicas estadísticas es analizar el grado de la relación existente entre variables utilizando modelos matemáticos y representaciones gráficas. Así pues, para representar la relación entre dos o más variables se dispone de una ecuación que permitirá estimar una variable en función de la otra.

A continuación, se estudia dicho grado de relación entre dos variables en lo que se conoce como *análisis de correlación*. Para representar esta relación se utiliza una representación gráfica llamada *diagrama de dispersión* y, finalmente, un modelo matemático para estimar el valor de una variable basándonos en el valor de otra, en lo que se denomina *análisis de regresión*.

3.3.1 Correlación Lineal.

En ocasiones nos puede interesar estudiar si existe o no algún tipo de relación entre dos variables aleatorias. El análisis de correlación no da la relación de la variable x con la variable y , nos da la linealidad; que es bueno cuando se acerca a 1 o a -1, o sea existe buena relación entre las variables.

Así, por ejemplo, podemos preguntarnos si hay alguna relación entre las notas de la asignatura Estadística I y las de Matemáticas I. Una primera aproximación al problema consistiría en dibujar en el plano R^2 un punto por cada alumno: la primera coordenada de cada punto sería su nota en estadística, mientras que la segunda sería su nota en matemáticas. Así, obtendríamos una nube de puntos la cual podría indicarnos visualmente la existencia o no de algún tipo de relación (lineal, parabólica, exponencial, etc.) entre ambas notas.

En particular, lo que interesa para este análisis es cuantificar la intensidad de la relación **lineal** entre dos variables. El parámetro que nos da tal cuantificación es el **coeficiente de correlación lineal de Pearson (r)**, cuyo valor oscila entre -1 y $+1$.

El valor de r se aproxima a $+1$ cuando la correlación tiende a ser lineal directa (mayores valores de X significan mayores valores de Y), y se aproxima a -1 cuando la correlación tiende a ser lineal inversa.

Es importante notar que la existencia de correlación entre variables no implica causalidad.

Si no hay correlación de ningún tipo entre dos variables, entonces tampoco habrá correlación lineal, por lo que $r = 0$. Sin embargo, el que ocurra $r = 0$ sólo nos dice que no hay correlación lineal, pero puede que la haya de otro tipo.

3.3.2 Definición y características del concepto de Regresión Lineal.

En aquellos casos en que el coeficiente de regresión lineal sea “cercano” a $+1$ o a -1 , tiene sentido considerar la ecuación de la recta que “mejor se ajuste” a la nube de puntos (recta de mínimos cuadrados). Uno de los principales usos de dicha recta será el de predecir o estimar los valores de Y que obtendríamos para distintos valores de X . Estos conceptos quedarán representados en lo que se denomina **diagrama de dispersión**.

3.3.2.1 Definición del Coeficiente de Determinación.

El coeficiente de determinación, nos da la confiabilidad del modelo matemático, que se está pronosticando.

Denominamos **coeficiente de determinación R2** como el coeficiente que nos indica el porcentaje del ajuste que se ha conseguido con el modelo, es decir el porcentaje de la variación de Y que se explica a través del modelo que se ha estimado, es decir a través del comportamiento de X. A mayor porcentaje mejor es nuestro modelo para predecir el comportamiento de la variable

También se puede entender este coeficiente de determinación como el porcentaje de varianza explicada por la recta de regresión y su valor siempre estará entre 0 y 1 y siempre es igual al cuadrado del coeficiente de correlación (r).

$$R2 = r^2$$

Es una medida de la proximidad o de ajuste de la recta de regresión a la nube de puntos.

También se le denomina *bondad del ajuste*.

1- $R2$ nos indica qué porcentaje de las variaciones no se explica a través del modelo de regresión, es como si fuera la varianza inexplicada que es la varianza de los residuos.

4. ANÁLISIS ESTADÍSTICOS DE LOS DATOS DEL TRÁFICO TCP/ IP DE LA RED DEL DEEE

4.1 Análisis del Comportamiento diario

En el análisis del comportamiento del flujo de tráfico TCP/IP de la red del DEEE se utilizaron 3 interfaces, las cuales están definidas como eth0, eth1 y eth2. Los datos se obtuvieron de la captura realizada en la semana del miércoles 30 de septiembre al viernes 9 de octubre del 2009, los cuales se capturaron mediante la herramienta de seguridad TCPdump, para luego ser reproducidos a través de TCPReplay, monitoreando y analizando las principales características con la ayuda del software FLUKE NETWORK

PROTOCOL EXPERTS (capítulo 2), y finalmente los datos obtenidos de la captura exportarlos al Excel para su análisis estadístico en el cual se centra este capítulo

A continuación se presenta el análisis por día de los diferentes eventos de acuerdo a cada interface. Es importante señalar que los datos fueron tomados, en la mañana y en la tarde, por lo que el comportamiento será presentado de esta manera.

4.1.1 Análisis de los Datos del Tráfico de la Red del DEEE.

En el siguiente cuadro se puede observar el tráfico que circulo por el servidor cuya dirección IP corresponde a la 10.1.2.8 de la interfaz eth0, este tráfico muestra la cantidad de bytes correspondientes a la capa de red y de transporte centrándose en los protocolos de mayor concurrencia como son ICMP, TCP y UDP, también se muestran los eventos (tiempo) en los cuales se capturó dicho tráfico. Con estos parámetros de tiempo y tamaño de paquetes, se hizo el análisis que permitió determinar las características que representan el comportamiento normal del tráfico de la red en análisis.

De esta manera se pudo establecer la cantidad de bytes por protocolo que circula en un determinado tiempo, con lo cual se pudo obtener parámetros que definen si el tráfico que circula en la red tiene anomalías que pueden tratarse de un ataque a la red.

Además se puede observar las direcciones IP de origen y destino involucradas en la transmisión de información, dato adicional que determina cuál o hacia que IPs está relacionada la mayor cantidad de tráfico, como ya se analizó con la ayuda del programa FLUKE NETWORK PROTOCOL EXPERT en el Capítulo 2.

FID	Elapsed [sec]	Size	Destination	Source	Summary
0	55.64692716	1438	190.152.129.111	10.1.28.3	HTTP Continuation Packet R Port=13323 Data
1	55.6469304	831	190.152.129.111	10.1.28.3	HTTP Continuation Packet R Port=13323 Data
2	55.64693304	66	10.1.28.3	190.152.129.111	TCP SP=13333 DP=80 SYN SEQ=3779748793 ACK=0 LEN=0 WS=65535 OPT={ MSS SACKperm }
3	55.6469346	66	190.152.129.111	10.1.28.3	TCP SP=80 DP=13333 SYN SEQ=102544960 ACK=3779748794 LEN=0 WS=5840 OPT={ MSS SACKperm }
4	55.64693596	64	10.1.28.3	190.152.129.111	TCP SP=13331 DP=80 SEQ=1346846099 ACK=104846485 LEN=0 WS=65535
5	55.64693736	114	SERVERDC1	10.1.28.5	NB-NAME C ID=33897 OP=Refresh QN=<01><02> _MSBROWSE _<02><1>
6	55.64695004	64	88.246.64.147	201.234.84.173	TCP SP=7918 DP=3739 SEQ=0 ACK=2071020860 LEN=0 WS=0 RST
7	55.64695152	64	10.1.28.3	190.152.129.111	TCP SP=13328 DP=80 FIN SEQ=1487138385 ACK=108698186 LEN=0 WS=64213
8	55.646953	64	190.152.129.111	10.1.28.3	TCP SP=80 DP=13328 SEQ=108698186 ACK=1487138386 LEN=0 WS=6432

Tabla. 5.1. Ejemplo Datos Capturados

4.1.2 Cálculo del Tamaño de la Muestra.

El número de paquetes capturados correspondientes al tráfico obtenido en la mañana del miércoles 30 de septiembre suman 42.395, por lo cual el análisis con toda esa cantidad de datos no es factible realizarlo ya sea por el tiempo o la velocidad que se necesitaría para procesar toda esa cantidad, por lo tanto con la ayuda de la técnica de muestreo se procedió a su interpretación.

Aplicando la siguiente fórmula para el cálculo del tamaño de la muestra para datos globales:

$$n = \frac{PQZ^2N}{E^2(N-1) + Z^2PQ}$$

Considerando que el error sea del 50%, la probabilidad de éxito sea del 50% y suponiendo que $p=q=0.5$ que es la opción más segura para obtener una muestra representativa, con la cual se puede interpretar la totalidad de los datos obtenidos, se tiene:

$$n = \frac{(0.5)^2 * (2.58)^2 * 42.395}{(0.05)^2 * (42395 - 1) + (2.58)^2 * (0.5)^2}$$

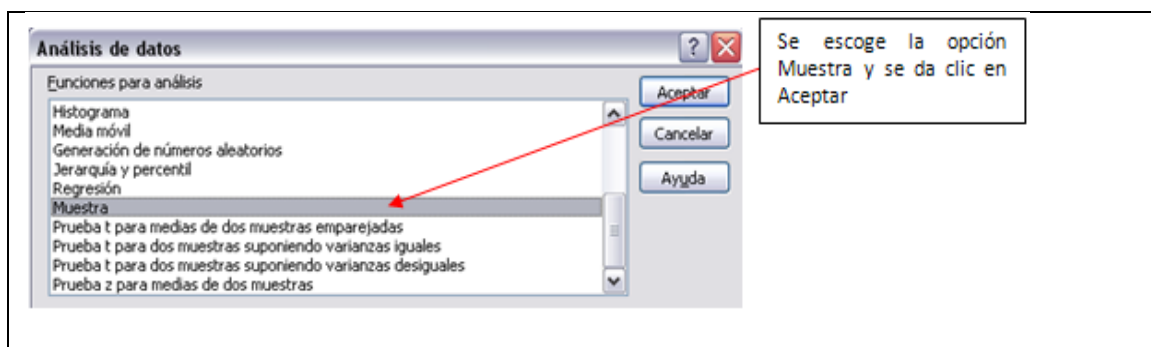
$$n = 655$$

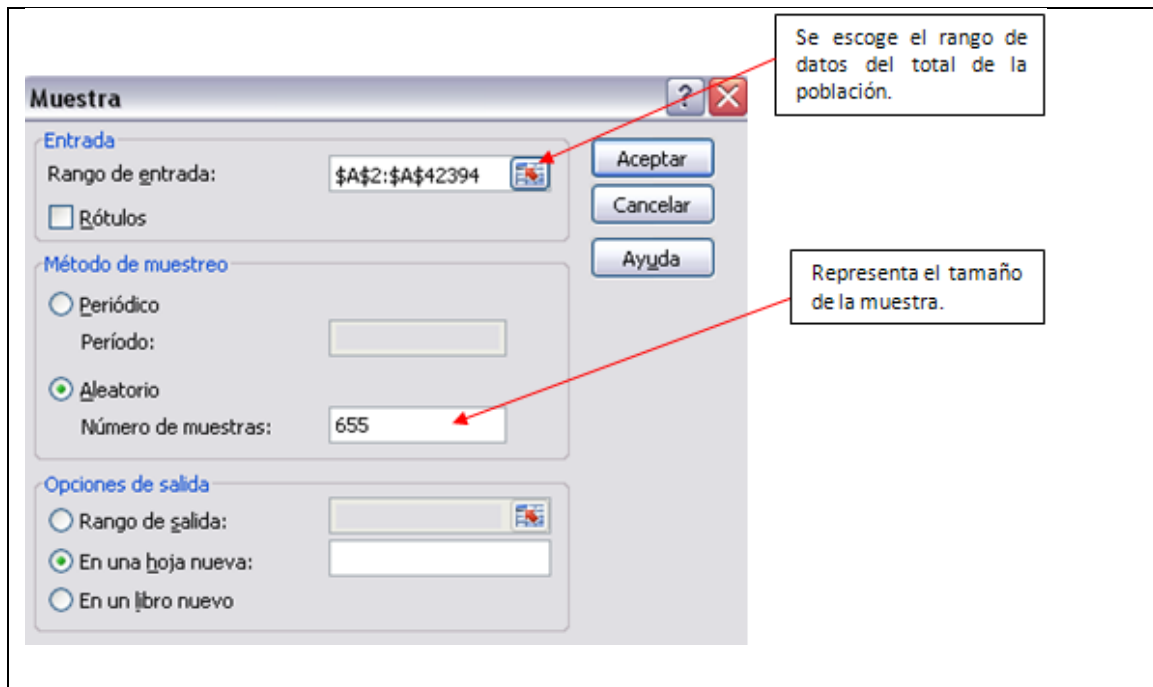
n =tamaño de la muestra
P =Probabilidad de éxito
$Q= 1 - P$ (probabilidad de fracaso)
N =Total de la población
Z^2 =Cte. Nivel Confianza
e^2 =Error

$n = 655$ representa el número de datos o muestras que se utilizó para la interpretación del tráfico que circuló por la interface eth0 de la red de pruebas TCP/IP correspondientes a la mañana del miércoles.

4.1.3 Aplicación de Muestreo (Herramientas / Análisis de Datos).

Para la aplicación de la técnica de muestreo mediante las herramientas de análisis de datos de EXCEL, se escoge en la opción Muestra, y a continuación se selecciona la totalidad de los datos (Población) de los cuales se desea obtener el número de muestras de acuerdo al tamaño calculado anteriormente, en este caso 655.





Se genera en una hoja nueva de Excel de las muestras aleatorias correspondientes a los datos que se desean analizar. En esta misma hoja de Excel con la ayuda de la función BUSCARV se extrae la información necesaria según las muestras obtenidas.

4.1.4 BUSCARV (Valor que se desea buscar en la matriz; Matriz de datos donde buscar datos; Columna que se desea obtener dato; Ordenado).

Excel busca en la primera columna de la matriz, definida en el segundo argumento, de forma vertical el valor que se coloca en el primer argumento.

Normalmente esta búsqueda Excel la hace pensando que esta primera columna está ordenada. Si los valores no lo estuvieran se debe especificar para que pueda encontrar el dato. Si la tabla no está ordenada se escribirá Falso en el argumento llamado Ordenado.

De esta manera se genera la tabla correspondiente a las muestras aleatorias obtenidas, la cual representa y caracteriza a la totalidad de los datos. Al ser todavía esta tabla extensa en cuanto a valores, únicamente se puede visualizar a continuación un ejemplo de algunos de los valores muestreados.

FID	Elapsed [se]	Size	Summary
74	55.6532714	1438	HTTP
107	55.65585872	82	TCP
204	55.65760556	138	TCP
205	55.6576074	202	TCP
225	55.6580358	138	TCP
416	55.66749508	82	ICMP
429	55.66753792	66	TCP
531	55.67478472	1518	TCP
570	55.67566668	64	TCP
589	55.67607568	64	TCP
672	55.68123784	1518	TCP
698	55.6825454	82	ICMP
711	55.6825888	82	TCP
756	55.68424652	82	TCP
902	55.69156464	64	TCP
1124	55.70574396	1438	HTTP
1342	55.71822832	64	TCP

Tabla. 5.2. Ejemplo de Muestras Obtenidas

4.1.5 Análisis: Miércoles mañana primera semana (eth0)

4.1.5.1 Cantidad de Bytes por segundo.

Se analizaron los datos que corresponden al periodo de captura de la mañana del día miércoles que duro desde las 8:00 am hasta las 12:00 pm momento en el cual se detuvo la captura, con el fin de extraer información sobre la cantidad de bytes que arriban por segundo.

La (Fig. 5.4.), muestra la distribución de las cantidades de bytes arribadas por segundo. De esta muestra, el promedio de bits arribados por segundo es

de 2389028, 2940 bps. Del gráfico se puede observar que la cantidad de bytes que arriban al canal por segundo es muy variable, esto es normal puesto que el tráfico de la red está afectado por una gran cantidad de factores, tales como: los servicios de la red, las horas pico de trabajo, y gran variedad de aplicaciones que corren en la red.

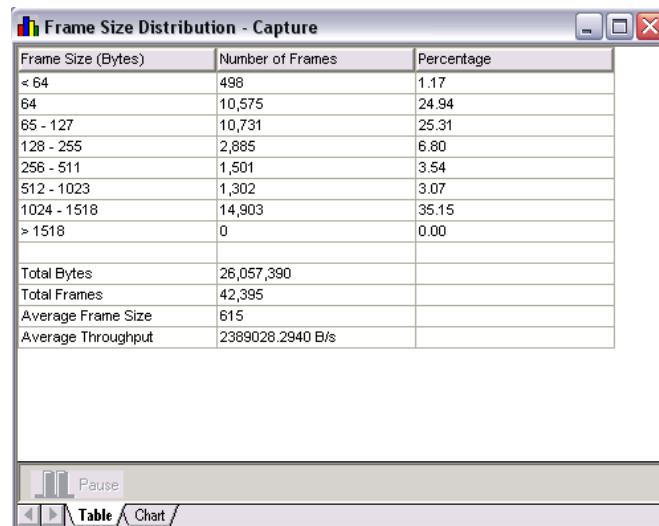


Figura. 5.4. Distribución de las cantidades de bytes arribadas por segundo

También se puede observar en la figura, que el tráfico llega con un patrón por ráfaga, es decir, en ciertos minutos se aumenta la cantidad de tramas que llegan y luego vuelve a la normalidad. Este patrón muestra que los usuarios tienden a usar la red intermitentemente, con tiempos de arribos entre tramas en general mucho mayor que el tiempo de transmisión del usuario. De acuerdo con Hammond and O'Reilly (1988), este tipo de variación de la carga, el cual parece típico de la mayoría de los tráficos de redes de computadoras, es llamado "bursty traffic". (Tráfico a ráfagas se refiere a un patrón desigual de la transmisión de datos: en algún momento la tasa de transmisión de datos es muy alta, mientras que otro momento podría ser muy baja.)

En la (Fig. 5.5.), se aprecia que la variación está en función de la cantidad de bits y no en el tiempo que transcurre en transferirse los datos, el cual se mantiene estable.

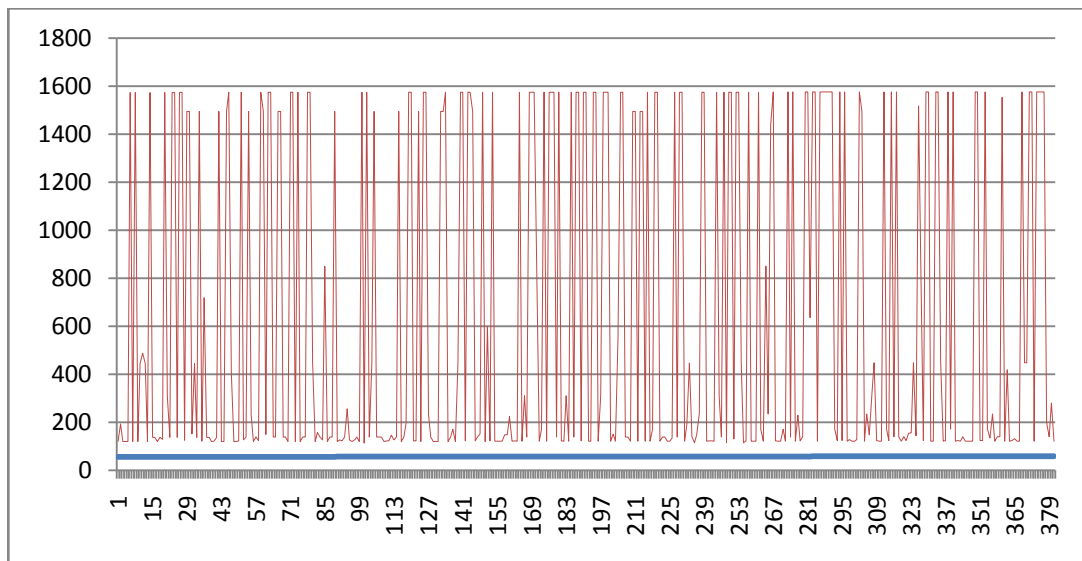


Figura 5.5. Variación en función de la cantidad de bites

4.1.5.2 Tiempo entre arribos de tramas.

Para obtener el tiempo promedio entre arribos se utilizó los datos del análisis de monitoreo obtenido a través del programa FLUKE NETWORK PROTOCOL EXPERT, con el cual se registra el tiempo entre arribo con respecto a la trama que ingresó al canal inmediatamente anterior a la trama analizada. El promedio de tiempo entre arribo fue de 0.428 mili segundos.

La (Figura. 5.6.), muestra el valor de los tiempos entre arribo de las 381 tramas observadas.

Server Name	Protocol	Minimum Time (ms)	Maximum Time (ms)	Average Time (ms)	Connections
10.1.0.101	DNS	0.001	28.81	0.988	252
10.1.0.104	DNS	0.001	25.352	1.01	110
10.1.28.3	SMTP	0.002	1.237	0.181	7
64.211.42.193	DNS	0.002	0.002	0.002	2
64.211.42.196	DNS	0.003	1.268	0.335	10
66.135.207.137	DNS	0.003	0.003	0.003	2
216.239.34.10	DNS	0.003	3.398	1.06	8
10.1.28.3	HTTP	0.004	4.989	0.423	16
74.125.95.9	DNS	0.005	0.025	0.012	8
192.48.79.30	DNS	0.005	0.008	0.006	4
65.55.226.140	DNS	0.006	1.231	0.619	4
4.71.104.165	DNS	0.007	0.423	0.215	4
192.5.5.241	DNS	0.008	0.017	0.013	4
192.41.162.30	DNS	0.008	0.347	0.177	8
204.2.249.36	DNS	0.008	0.019	0.014	4
70.42.23.252	DNS	0.013	1.275	0.569	6
67.106.145.253	DNS	0.014	0.014	0.014	2
74.125.77.9	DNS	0.014	4.263	1.176	8
204.2.249.46	DNS	0.014	0.017	0.016	4
201.37.147.73	HTTPS	0.015	2.969	0.538	16
204.2.249.38	DNS	0.016	0.02	0.018	4
204.2.249.5	DNS	0.016	0.016	0.016	2
64.215.156.30	DNS	0.017	0.017	0.017	2
192.26.92.30	DNS	0.017	3.405	0.963	8
203.105.65.195	DNS	0.017	0.017	0.017	2
66.218.167.2	DNS	0.018	0.018	0.018	2
192.31.80.30	DNS	0.018	0.415	0.151	6
66.218.167.3	DNS	0.018	0.018	0.018	2
66.135.215.5	DNS	0.019	0.019	0.019	2
192.42.93.30	DNS	0.019	0.019	0.019	2
66.135.207.138	DNS	0.019	0.019	0.019	2
204.2.249.6	DNS	0.02	3.447	1.165	6
65.203.229.15	DNS	0.02	0.02	0.02	2
128.242.118.131	DNS	0.02	1.312	0.588	6
17.112.144.50	DNS	0.021	0.021	0.021	2
204.2.178.133	DNS	0.021	0.858	0.417	6
216.239.38.10	DNS	0.024	0.851	0.568	6
209.59.13.227	DNS	0.025	0.025	0.025	2
216.239.36.10	DNS	0.027	0.861	0.444	4
4.23.59.51	DNS	0.031	0.031	0.031	2
192.12.94.30	DNS	0.038	2.993	1.137	6
62.41.78.201	DNS	0.331	0.331	0.331	2
192.43.172.30	DNS	0.335	0.335	0.335	2
17.112.144.59	DNS	0.35	0.35	0.35	2
17.254.0.59	DNS	0.361	0.361	0.361	2
208.44.108.138	DNS	0.372	0.372	0.372	2
91.200.16.100	DNS	0.393	0.393	0.393	2
66.218.167.1	DNS	0.394	0.394	0.394	2
77.72.229.252	DNS	0.413	0.413	0.413	2
192.112.36.4	DNS	0.425	0.425	0.425	2
195.59.44.134	DNS	0.426	0.426	0.426	2
81.7.134.249	DNS	0.444	0.444	0.444	2
12.1.41.132	DNS	0.448	0.448	0.448	2
12.1.41.134	DNS	0.808	0.808	0.808	2
81.17.254.6	DNS	0.851	0.851	0.851	2
208.80.125.2	DNS	1.691	1.691	1.691	2
78.153.212.176	DNS	4.616	4.616	4.616	2
		Tiempo Mínimo (ms)	Tiempo Máximo (ms)	Promedio (ms)	Conexiones
		0.220883333	1.712416667	0.428966667	590

Figura. 5.6. Gráfico de los tiempos entre arribo de las 381 tramas observadas.

Se puede observar de la tabla anterior que una gran cantidad de tiempos entre arribos son valores muy pequeños, la mayoría de los tiempos entre arribos son valores menores a 0,002 mili segundos, constituyéndose el 76% del total de los tiempos entre arribos registrados, esto es, el 76% de las

tramas son seguidas por la próxima trama dentro de menos de 2 milisegundos.

El estudio de los tiempos entre arribos es importante porque a través de esta información se puede analizar el intervalo de tiempo que transcurre entre un arribo de paquete al canal y el siguiente, lo que permitiría analizar a su vez la intensidad de arribos de éstos al canal. Al tener la mayoría de los tiempos entre arribos valores muy pequeños, significa que están llegando paquetes al canal en forma muy seguida

Además se debe cuantificar la cantidad de bits que arriban al canal de acuerdo a los protocolos que conforman las capas de Red (IP, ICMP) y de Transporte (TCP, UDP) para así tener una relación del porcentaje que cada uno de ellos ocupa dentro del tráfico total capturado. Esta interpretación es importante a la hora de determinar el modelo del tráfico de red ya que lo caracteriza de mejor manera, permitiendo al administrador de red saber qué tipo de ataque se está produciendo y analizar independientemente el tráfico de acuerdo a cada protocolo, pues a más de obtener un modelo que determine el comportamiento de la totalidad del tráfico, es importante considerar que se puede producir un instante en el cual por condiciones normales baje considerablemente el porcentaje de transferencia de datos relacionado a cualquier tipo de protocolo, momento en el cual se puede originar un ataque que vulnere la configuración de otro, nivelando así el porcentaje normal de utilización de la red, siendo incapaz el sistema de determinar a tiempo el ataque. Por este motivo si se analiza primero el tráfico de manera general con todos sus protocolos involucrados se puede tener las características que rigen al comportamiento normal del tráfico que circula por la red, pero se debe considerar el porcentaje de cada uno para tener un parámetro adicional que determine la excepción descrita.

A continuación se describe la cantidad de bytes por protocolo relacionado a la interface eth0.

Suma de Size		
Summary ▼	Total	Cantidad
802.1	673500	2400
ARP	8512	133
BPDU	28386	498
DNS	204397	1615
DNS F	1301	1
HTTP	1531408	1149
HTTPS	7664	38
ICMP	917464	3893
Kerbe	10797	8
LDAP	228	2
LDAP:	2458	12
NB-NA	92238	817
NetBI	3868	14
NETCP	342	3
NTP	684	6
Post	574	6
SLP V	342	3
SMB Q	45348	239
SMB R	56028	231
SMB:	225724	154
SMTP	805264	611
SNMP	3936	32
Sybas	64	1
TCP	19746743	29012
TCP F	55981	194
TCP R	1344	21
TCP W	1621851	1131
UDP	10944	171
Total general	26057390	42395

Figura. 5.7. Porcentaje Tráfico de cada protocolo

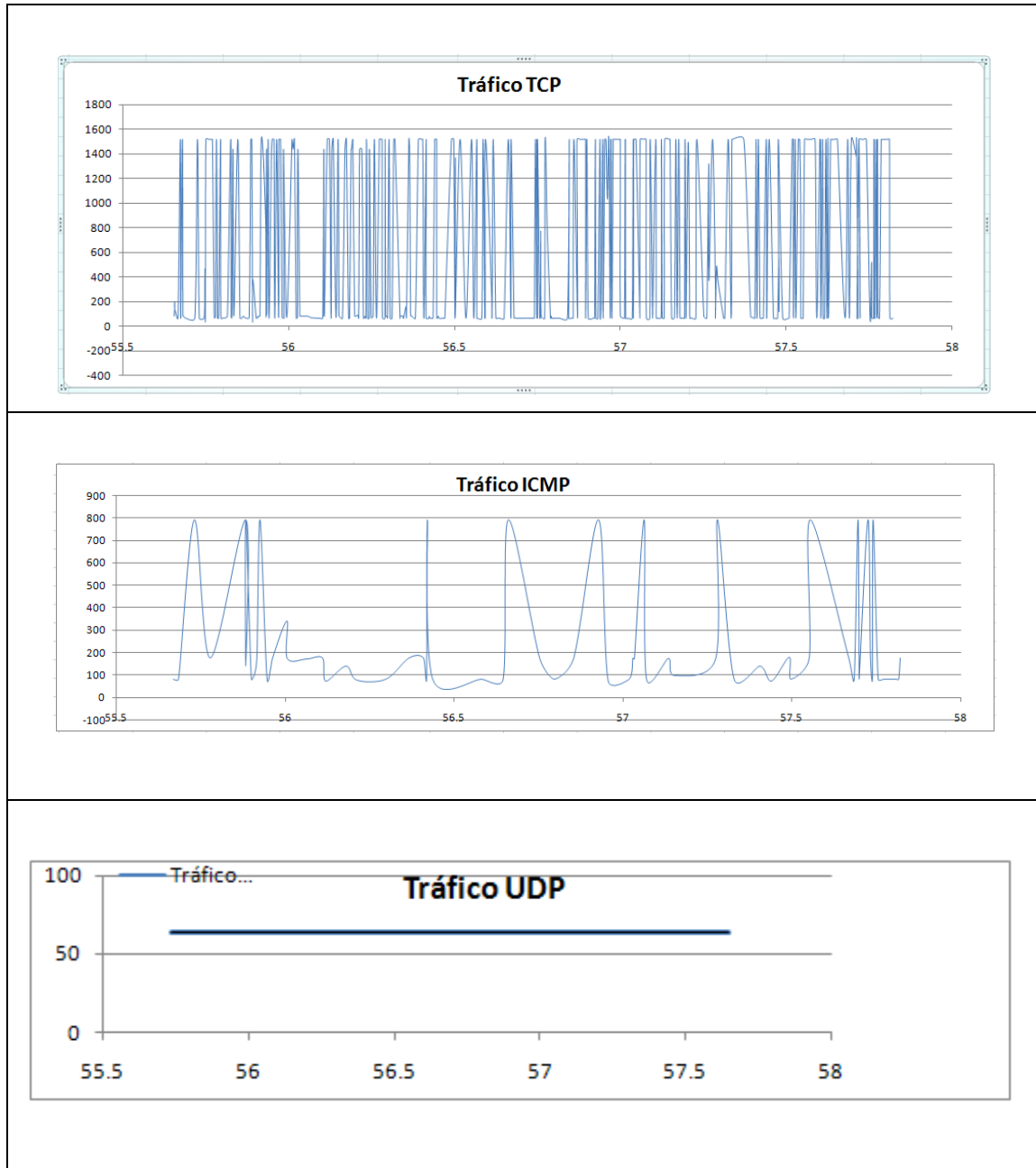


Figura 5.8. Trafico TCP, ICMP, UDP

Como se puede observar el mayor porcentaje corresponde al tráfico TCP, con un 75% del total, determinando así que es muy difícil que este porcentaje dentro de la red baje a porcentajes menores e iguales al de los otros, sin embargo ya que como se verá más adelante el trafico de una red es muy variable, es de importancia por lo anterior mencionado el conocer el

porcentaje de cada protocolo y establecer umbrales mínimos y máximos del $\pm 15\%$, para las 3 interfaces.

El porcentaje para el protocolo ICMP corresponde al 5 %, y el porcentaje del protocolo UDP 0.4 %

4.1.5.3 Desarrollo del Modelo de Tráfico.

Una vez capturados los datos se analiza la estructura de correlación que presenta la muestra, de aquí se propone y se estima un modelo basado en ecuaciones con las cuales se busca capturar la dinámica de la serie, de ser correcta la formulación del modelo, este se valida y después se pronostican las futuras observaciones.

4.1.5.4 Extracción de la Serie.

Para el desarrollo del modelo de tráfico de la Red del DEEE se utilizó la muestra de datos obtenida del tráfico capturado. Como lo han confirmado varios estudios de tráfico en redes, el tráfico actual como el de Internet e incluso el de video, presenta características de correlación.

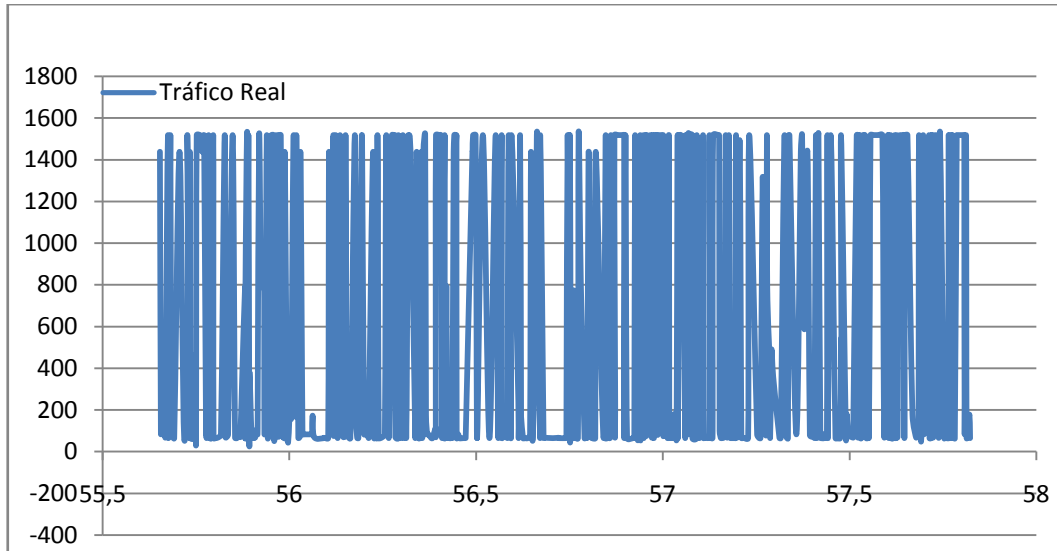


Figura. 5.9. Serie de Tiempo del Tráfico Capturado

Los datos de tráfico fueron extraídos a través de la herramienta TCPDUMP, y analizados mediante FLUKE NETWORKS PROTOCOL EXPERT para luego exportarlos a EXCEL para su interpretación estadística.

Estos datos se capturaron como una variable de paquetes por segundo y se tomaron por cada día muestras que corresponden a una semana de captura.

En la figura 5.9 se puede observar la serie de tráfico obtenida para los 655 datos capturados correspondientes a la interface eth0 de la mañana del miércoles 30 de septiembre del 2009.

4.1.6 Identificación del Modelo

4.1.6.1 Definición y características del concepto de Regresión Lineal.

A partir del concepto de Regresión Lineal, se procede a analizar la relación entre ambas variables, la variable Cantidad de bits (c) que es la variable dependiente del modelo y la variable que vamos a analizar y el Tiempo (t) que es la variable independiente o la variable explicativa que se utiliza para estudiar el comportamiento normal del tráfico de la Red del DEEE.

En este modelo se desea comprobar qué influencia tiene el Tiempo sobre la Cantidad de bits que circulan por la red.

Para poder cuantificar dicha relación, se debe también representar la recta de regresión que subyace en el modelo matemático que relaciona ambas variables.

La relación entre ambas variables y una aproximación de la magnitud de la influencia del Tiempo sobre la Cantidad de bits que circulan por la red se estima utilizando el modelo por mínimos cuadrados ordinarios (M.C.O.) donde se minimiza la suma de los cuadrados de los residuos.

La recta en negro (que aparece a continuación en el gráfico), es la que mejor se ajusta a la nube de puntos que tenemos. Dicho de otra forma, es la recta que hace que el error de estimación, definido como la distancia entre el valor observado y el valor estimado de la variable endógena, sea la mínima para cada una de las observaciones (recta de mínimos cuadrados), esta recta será la que se utiliza para predecir o estimar los valores de Y que se obtienen para distintos valores de X.

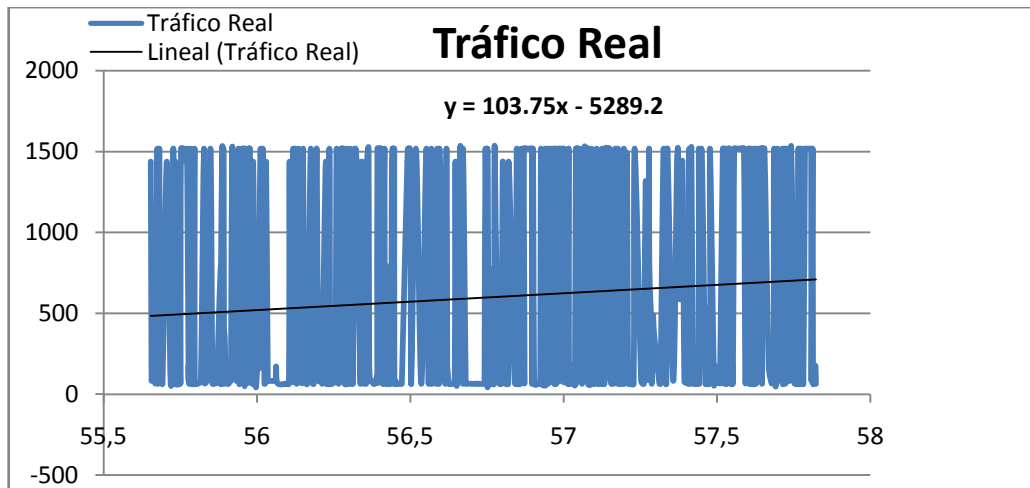


Figura. 5.10. Tráfico Real

La diferencia entre un valor observado y el valor estimado lo denominaremos **residuo**.

$$Residuo = t t Y - Y'$$

El problema consiste en minimizar la suma de los cuadrados de los residuos.

Del gráfico, cabe observar que no todos los puntos están en la línea de regresión.

Si todos lo estuvieran y, además, si el número de observaciones fuera suficientemente grande, no habría ningún error de estimación. En ese caso, no habría ninguna diferencia entre el valor observado y el valor de predicción.

Como imaginamos, en los casos reales, las predicciones perfectas son prácticamente imposibles y lo que necesitamos es una medida que describa cómo de precisa es la predicción de Y en función de X o, inversamente, qué inexacta puede ser la estimación.

A esta medida se le llama error estándar de estimación y se denota S_{yx} . El error estándar de estimación, es el mismo concepto que la desviación estándar, aunque ésta mide la dispersión alrededor de la media y el error estándar mide la dispersión alrededor de la línea de regresión.

4.1.6.2 Interpretación de los coeficientes estimados.

En la tabla de los valores el **Pronóstico para y** (documento adjunto Excel) significan los valores obtenidos con la ecuación anterior, los **residuos**, son la diferencia que hay entre los históricos (el evento real) y los valores pronosticados, los cuales se grafican, donde se aprecian el mejor caso y peor caso, o sea mi mejor pronóstico y mi peor pronóstico, los más alejados de la recta son el mayor error en mi pronóstico.

O sea cuando x (el tiempo) vale 40,68 el tamaño que circula es de 224,15, este es el error que voy a cometer, que puede ser mayor o menor en dependencia de la cercanía que se encuentre de la recta.

Y cuando no haya variación en la cantidad de bits, el tiempo según la recta será negativo.

La **correlación** entre ambas variables es alta, ya que el coeficiente de correlación $r = 0.87$ está muy próximo a 1.

En nuestro trabajo, el **coeficiente de determinación** nos da bajo, el 75,3%, por lo que sólo conseguimos explicar el 75,3 % de las variaciones de las cantidades de bits a través del ajuste por medio del Tiempo.

4.1.6.3 Inferencia en el modelo de regresión.

Una vez que hemos calculado la recta de regresión y el ajuste que hemos conseguido con el modelo de regresión, el siguiente paso consiste en analizar si la regresión en efecto es válida y la podemos utilizar para predecir. Para ello debemos contrastar si la correlación entre ambas variables es distinta de cero o si el modelo de regresión es válido en el sentido de contrastar si el análisis de nuestra variable endógena (Y) es válido a través de la influencia de la variable explicativa (X).

Supongamos por un lado que el coeficiente de correlación r , está próximo a +1 o a -1, y por tanto parece indicar la existencia de una correlación entre los valores de la muestra. Pero este valor del coeficiente de correlación muestral entre ambas variables no garantiza que también estén correlacionadas en la población.

Ahora lo que se debe es comprobar si esta estimación de este modelo es válida en el sentido de si es significativa de forma que la variable Tiempo (X) es relevante para explicar (Y) que es la cantidad de bits. Entonces debemos contrastar si la **pendiente de la recta de regresión poblacional** 2β es significativamente distinta de cero, de ahí tendríamos que, en efecto, existe una correlación entre ambas variables poblacionales.

En el análisis realizado el tiempo es la variable independiente, o sea independientemente del tiempo, van a depender el tamaño de cada tipo de protocolo que circule por la red, con una ecuación pronóstico $y=ax+b$, donde b es la ordenada al origen y a es la pendiente de la recta.

Para el análisis estadístico vamos a análisis de datos, regresión, seleccionamos y como el tamaño y x como el tiempo, y procedemos a calcular.

Ahora para establecer el modelo matemático que voy a desarrollar para pronosticar, planteo la ecuación,

$$\text{Size} = (\text{Variable X } 1) * (\text{Tiempo}) + \text{Intercepción}$$

A y continuación presentamos el análisis de los diferentes estadígrafos:

<i>Elapsed [sec]</i>		<i>Size</i>	
Media	56,75903417	Media	582,086614
Error típico	0,032288344	Error típico	33,8769286
Mediana	56,752477	Mediana	90
Moda	56,62038388	Moda	1518
Desviación estándar	0,630243337	Desviación estándar	661,251266
Varianza de la muestra	0,397206664	Varianza de la muestra	437253,237
	-		-
Curtosis	1,131671269	Curtosis	1,53212217
Coficiente de asimetría	0,025429856	Coficiente de asimetría	0,64512575
Rango	2,16763456	Rango	1461
Mínimo	55,65499632	Mínimo	57
Máximo	57,82263088	Máximo	1518
Suma	21625,19	Suma	221775
Cuenta	381	Cuenta	381
Nivel de confianza(99,0%)	0,063486192	Nivel de confianza(99,0%)	66,6097088

Tabla. 5.3. Análisis de los Diferentes Estadígrafos

4.1.6.4 Análisis de regresión

<i>Estadísticas de la regresión</i>	
Coefficiente de correlación múltiple	0,873152283
Coefficiente de determinación R ²	0,75360282
R ² ajustado	0,742108056
Error típico	0,172917636
Observaciones	380

Tabla. 5.4. Estadísticas de la Regresión

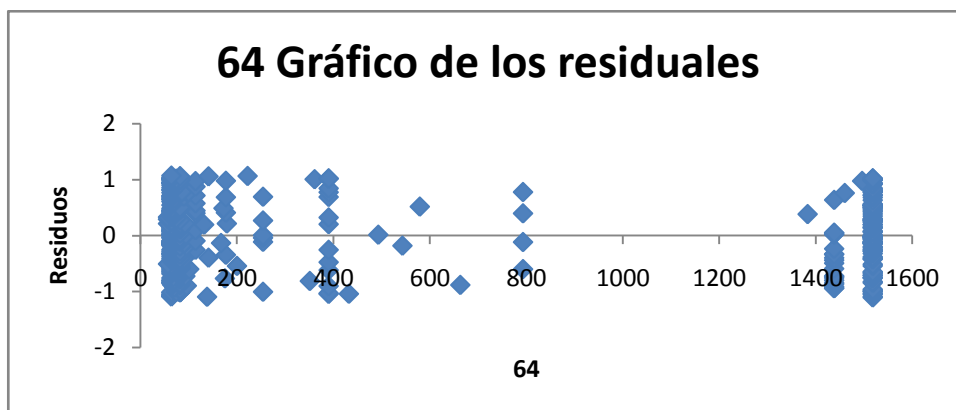


Figura. 5.11. Gráfico de los Residuales

4.1.7 Análisis: Miércoles tarde, primera semana (eth0)

4.1.7.1 Comportamiento de la variable y

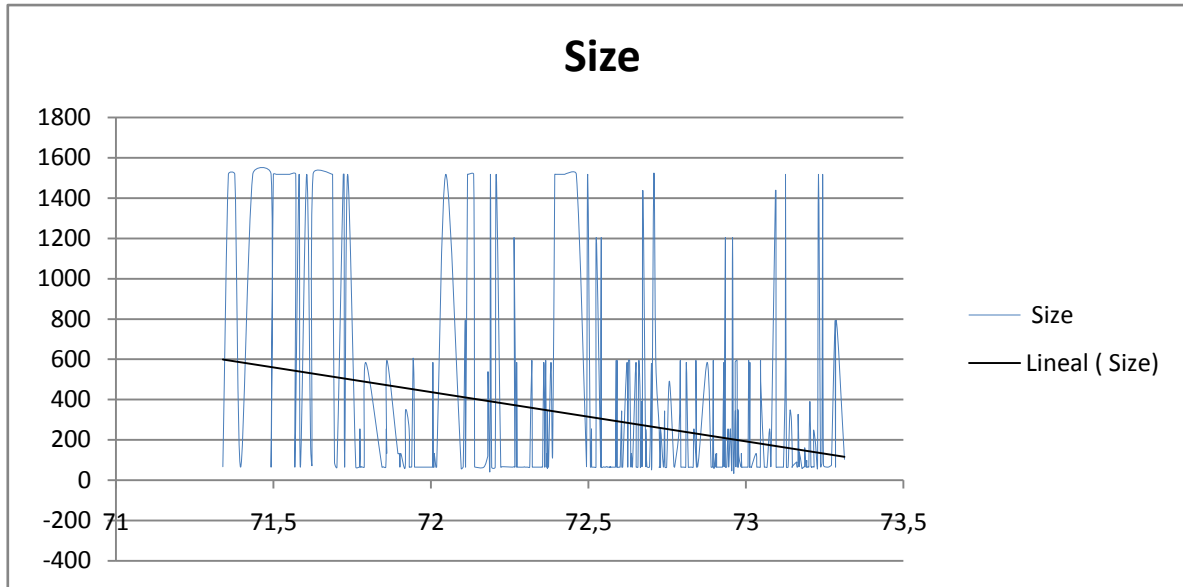


Figura. 5.12. Comportamiento de la Variable “y”

4.1.7.2 Comportamiento de ambas variables

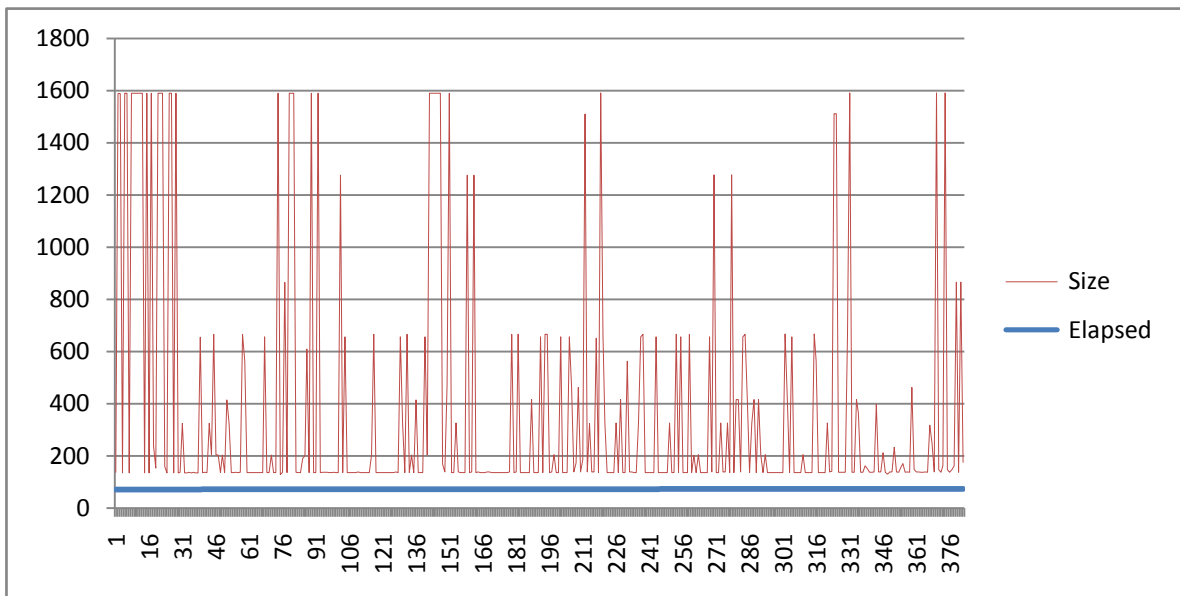


Figura. 5.13. Comportamiento de Ambas Variables

4.1.7.3 Análisis estadígrafos.

<i>Elapsed</i>		<i>Size</i>	
Media	72,5563017	Media	301,068063
Error típico	0,025144324	Error típico	23,1002209
Mediana	72,61597722	Mediana	66
Moda	72,55	Moda	64
Desviación estándar	0,4914413	Desviación estándar	451,489667
Varianza de la muestra	0,241514551	Varianza de la muestra	203842,919
Curtosis	0,696623395	Curtosis	2,51223523
Coficiente de asimetría	0,488849301	Coficiente de asimetría	1,97708343
Rango	1,97327424	Rango	1461
Mínimo	71,33971156	Mínimo	57
Máximo	73,3129858	Máximo	1518
Suma	27716,50725	Suma	115008
Cuenta	382	Cuenta	382
Nivel de confianza(95,0%)	0,049439018	Nivel de confianza(95,0%)	45,4198816

Tabla.5.5. Análisis de los Diferentes Estadígrafos

4.1.7.4 Análisis de regresión.

La aplicación estadística correspondiente a la regresión calculada mediante herramientas de Excel aplicadas a los datos obtenidos como muestras de la totalidad del tráfico, permiten asignar valores aleatorios de la serie que modela el comportamiento normal del tráfico y mediante el gráfico de los residuales pronosticar posibles comportamientos anómalos de la red, los

cuales pueden tratarse de ataques a la red o simplemente son el producto de un tráfico lícito generado por los usuarios. Con lo explicado en el capítulo 3 se puede interpretar este tipo de tráfico en tiempo real, determinar si se está utilizando una técnica de ataque y tomar los correctivos necesarios.

<i>Estadísticas de la regresión</i>	
Coeficiente de correlación múltiple	0,82663466
Coeficiente de determinación R^2	0,80709405
R^2 ajustado	0,78495627
Error típico	0,21752878
Observaciones	382

TABLA. 5.6. Estadística de la Regresión

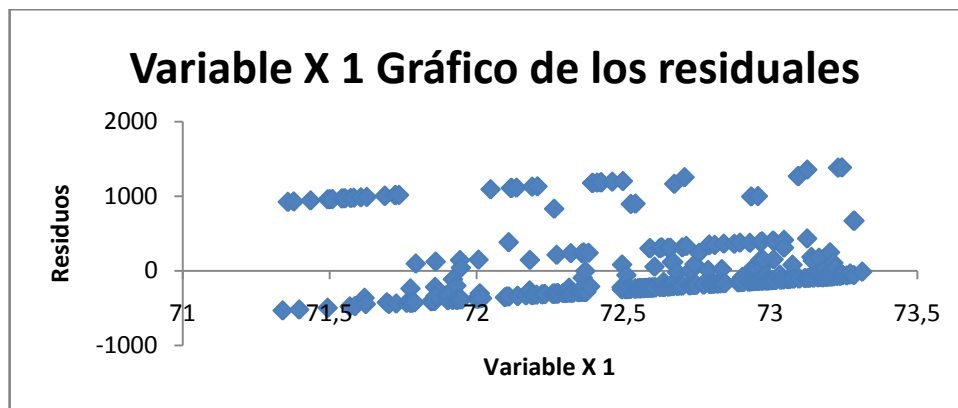


Figura. 5.14. Variable X1 Gráfico de los residuos

Puede apreciarse que el flujo de información en la red ha disminuido en el día, en la tarde circularon menor bits que en la mañana, la tendencia en la tarde es a disminuir, la correlación entre ambas variables es alta sobrepasa el 85 %, la cantidad mínima de bits que circularon fueron de 57 y el máximo de 1518 bits.

En la mañana la mayoría de los paquetes que circularon en la red eran de 1518 bits, y en la tarde de 64 bits, lo que nos supone que el trabajo en la mañana es más intenso en cantidad de paquetes y número de bits.

4.1.8 Análisis: Miércoles mañana segunda semana (eth 0)

4.1.8.1 Comportamiento de la variable y

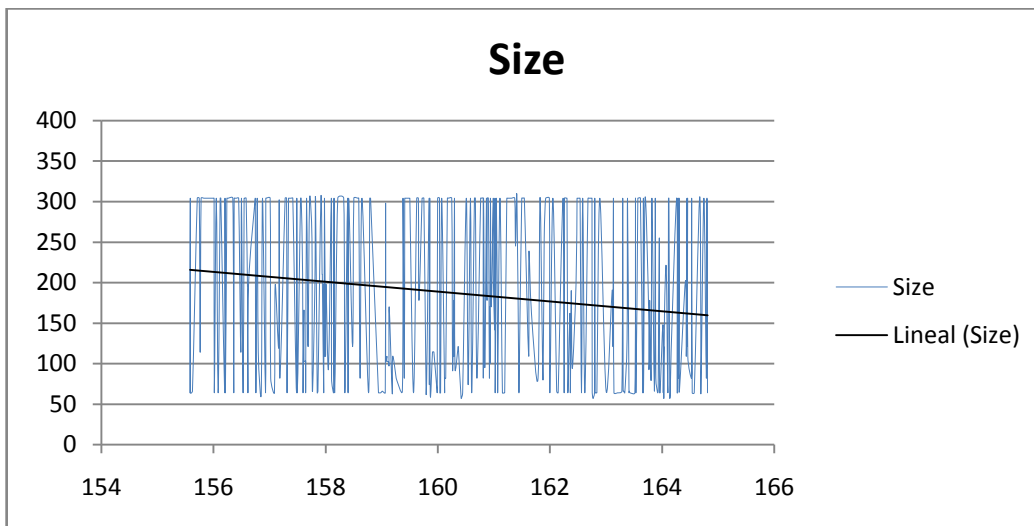


Figura. 5.15. Comportamiento de la Variable y

4.1.8.2 Comportamiento ambas variables

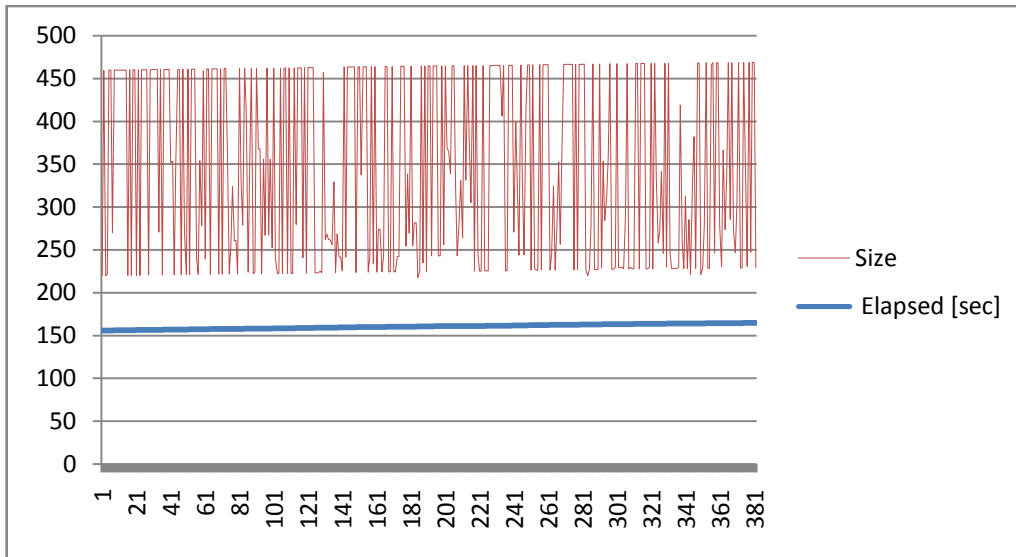


Figura. 5.16. Comportamiento de Ambas Variables

4.1.8.3 Análisis estadígrafos

<i>Elapsed [sec]</i>		<i>Size</i>	
Media	160,4391	Media	186,185864
Error típico	0,13884678	Error típico	0,150554905
Mediana	160,664831	Mediana	170
Moda	160,66	Moda	304
Desviación estándar	2,71373537	Desviación estándar	107,604967
Varianza de la muestra	7,36435963	Varianza de la muestra	11578,8289
Curtosis	1,22722474	Curtosis	-1,83658575
Coefficiente de asimetría	0,09955063	Coefficiente de asimetría	0,03873289
Rango	9,23144524	Rango	247
Mínimo	155,581562	Mínimo	57

Máximo	164,813007	Máximo	304
Suma	61287,7361	Suma	71123
Cuenta	382	Cuenta	382
Nivel de confianza(95,0%)	0,2730019	Nivel de confianza(95,0%)	10,8250647

Tabla. 5.7. Análisis Estadígrafos

4.1.8.4 Análisis de regresión

<i>Estadísticas de la regresión</i>	
Coeficiente de correlación múltiple	0,848819173
Coeficiente de determinación R ²	0,822147146
R ² ajustado	0,81956706
Error típico	0,250595955
Observaciones	381

Tabla. 5.8. Análisis de Regresión

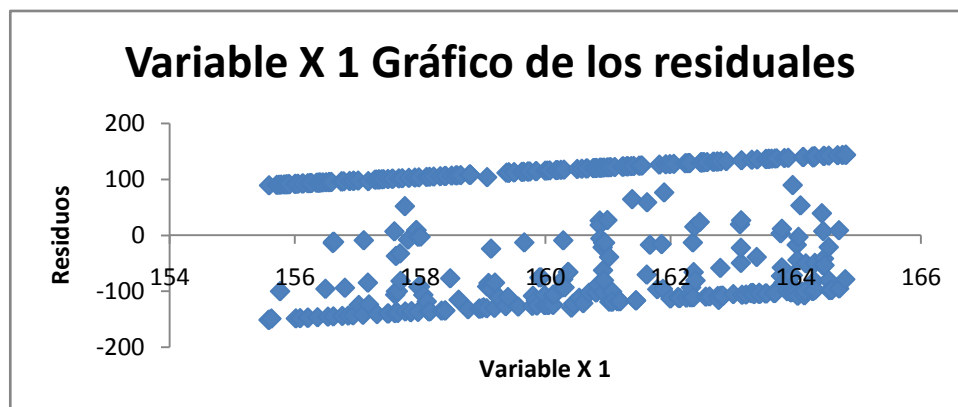


Figura. 5.17 Variable X1 Gráfico de los Residuales

4.1.9 Análisis: Miércoles tarde, segunda semana (eth 0)

4.1.9.1 Comportamiento de la variable y

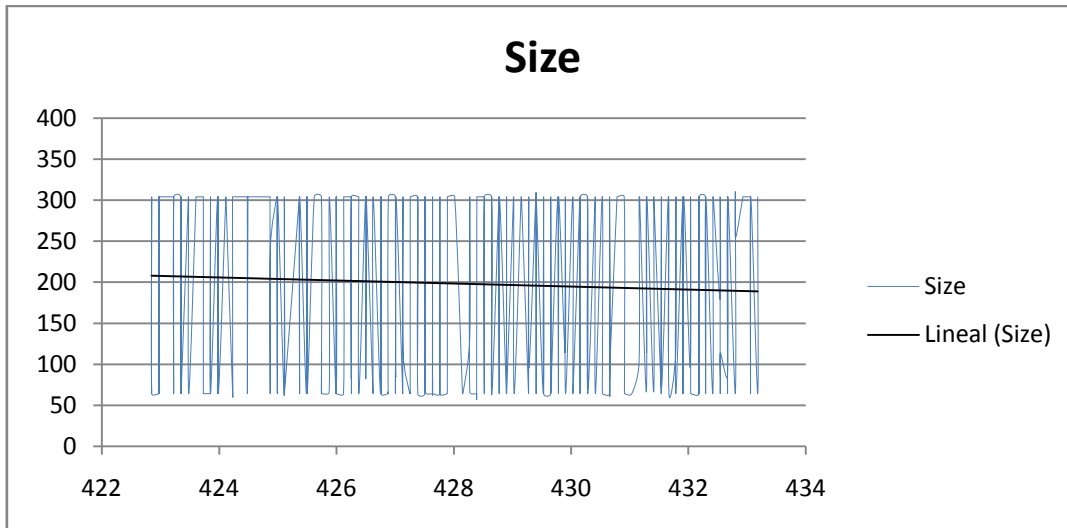


Figura. 5.18. Comportamiento de la Variable y

4.1.9.2 Comportamiento de ambas variables

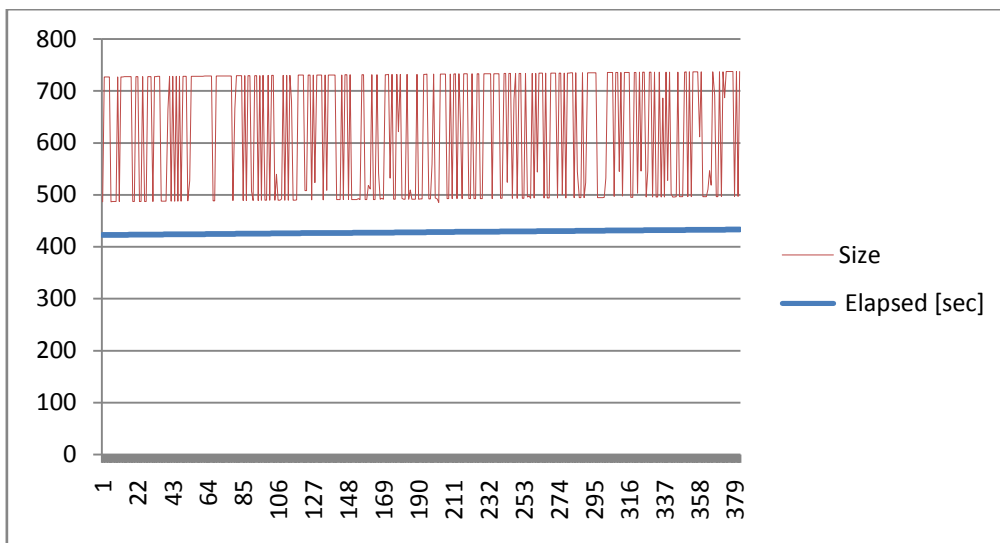


Figura. 5.19. Comportamiento de Ambas Variables

4.1.9.3 Análisis estadígrafos

<i>Elapsed [sec]</i>		<i>Size</i>	
Media	427,969032	Media	198,374346
Error típico	0,15457031	Error típico	0,19353677
Mediana	427,889058	Mediana	304
Moda	424,482792	Moda	304
Desviación estándar	3,02104902	Desviación estándar	116,005696
Varianza de la muestra	9,12673717	Varianza de la muestra	13457,3214
Curtosis	-1,2015809	Curtosis	1,92397705
Coficiente de asimetría	0,00725391	Coficiente de asimetría	0,22528711
Rango	10,3329592	Rango	247
Mínimo	422,848062	Mínimo	57
Máximo	433,181021	Máximo	304
Suma	163484,17	Suma	75779
Cuenta	382	Cuenta	382
Nivel de confianza(95,0%)	0,30391767	Nivel de confianza(95,0%)	11,6701784

Tabla. 5.9. Análisis Estadígrafos

4.1.9.4 Análisis de regresión

<i>Estadísticas de la regresión</i>	
Coeficiente de correlación múltiple	0,853198289
Coeficiente de determinación R ²	0,828300579
R ² ajustado	0,819900268
Error típico	0,199086643
Observaciones	381

Tabla. 5.10. Análisis de regresión

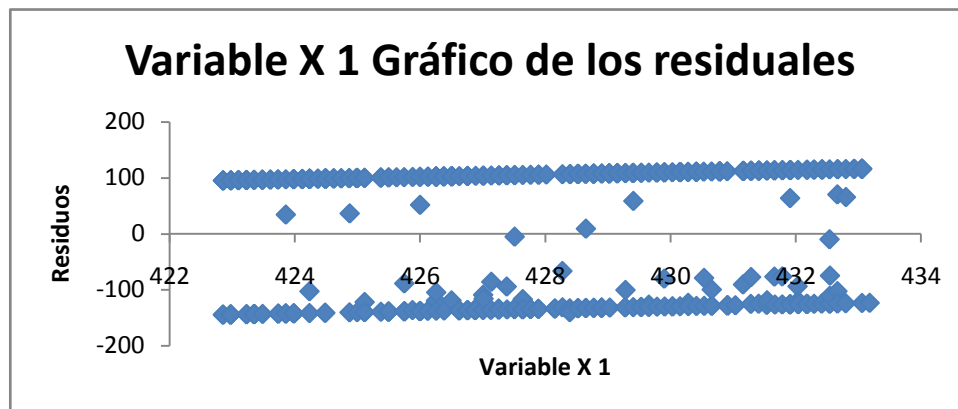


Figura. 5.20. Variable X1 Gráfico de los residuales

Puede apreciarse que el flujo de información en la red ha mantenido en el día, en la tarde circularon la misma cantidad de bits que en la mañana, dado por el día de la semana, la correlación entre ambas variables es alta sobrepasa el 82 %, la cantidad mínima de bits que circularon fueron de 57 y el máximo de 304 bits.

En la mañana la mayoría de los paquetes que circularon en la red eran de 304 bits, este análisis se realizó para cada día de la semana, los cuales aparecen en los anexos.

4.2 Análisis comportamiento todo el período

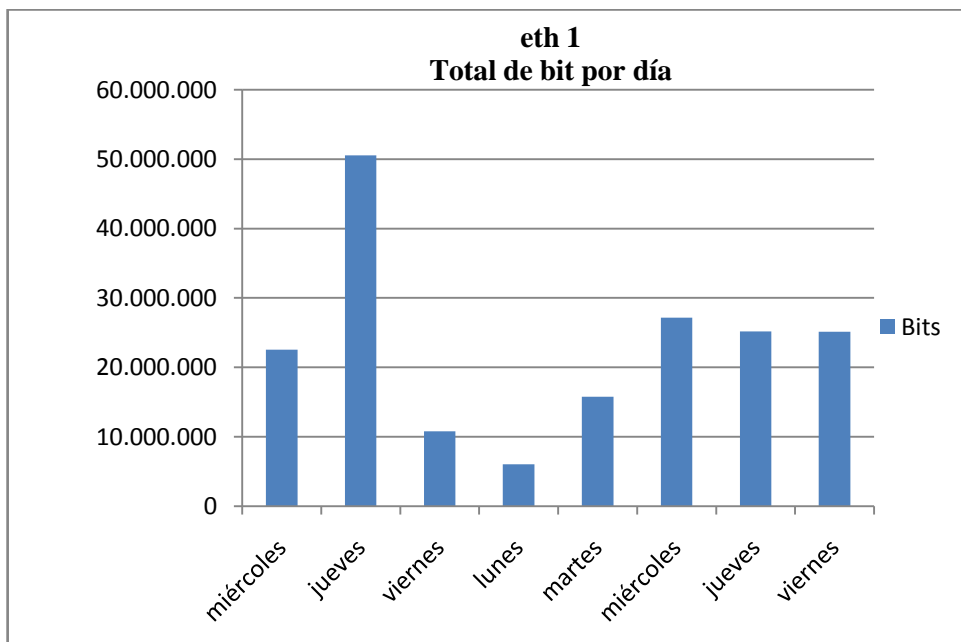
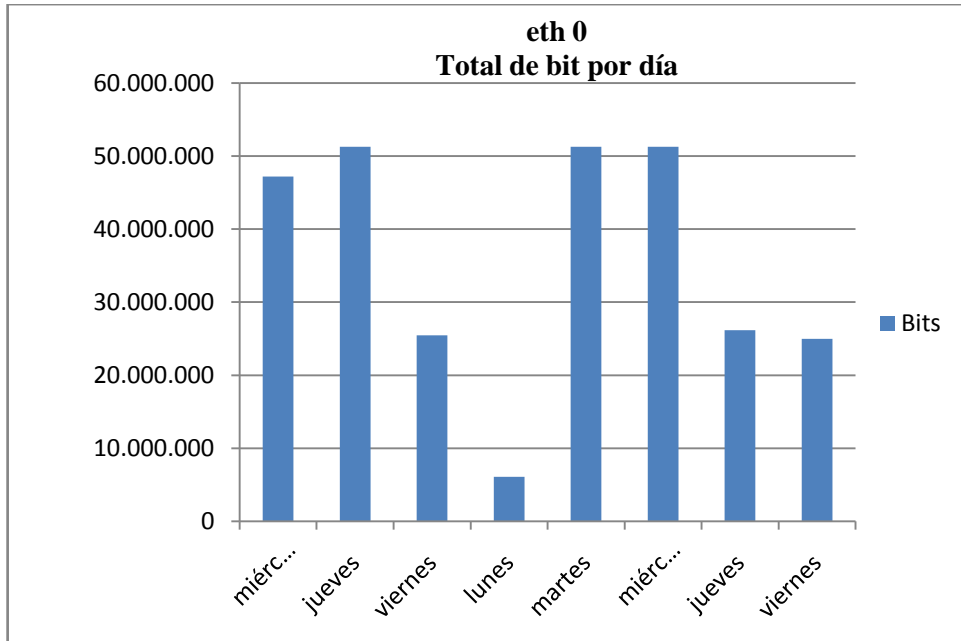
El análisis de todo el período se realizó atendiendo a la cantidad de bits que circularon en la red en los 8 días que se realizó el estudio, a continuación la tabla que recoge la cantidad de bits por día.

4.2.1 Cantidad de bits

DIAS	eth0			eth 1			eth 2			TOTAL
	mañana	tarde	día	mañana	tarde	día	mañana	tarde	día	
miércoles	26.057.390	21.136.708	47.194.098	22.557.472		22.557.472				69.751.570
jueves	25.043.162	26.247.508	51.290.670	24.685.180	25.840.912	50.526.092	24.763.174	25.477.662	50.240.836	152.057.598
viernes			25.469.313			10.774.387			25.585.927	61.829.627
lunes			6.086.967			6.054.402			5.909.510	18.050.879
martes	5.763.057	9.894.521	51.290.670	5.788.783	9.981.803	15.770.586	5.590.351	24.182.213	29.772.564	96.833.820
miércoles	11.593.050	13.130.167	51.290.670	13.630.073	13.540.225	27.170.298	13.916.219	13.695.154	27.611.373	106.072.341
jueves			26.160.228			25.190.473			25.303.598	76.654.299
viernes			25.000.973			25.121.356			25.503.572	75.625.901

Tabla. 5.11. Cantidad de bits que circularon en la red en los 8 días

A continuación presentamos el comportamiento de la cantidad de paquetes que circularon en la red por días, por cada interfaz.



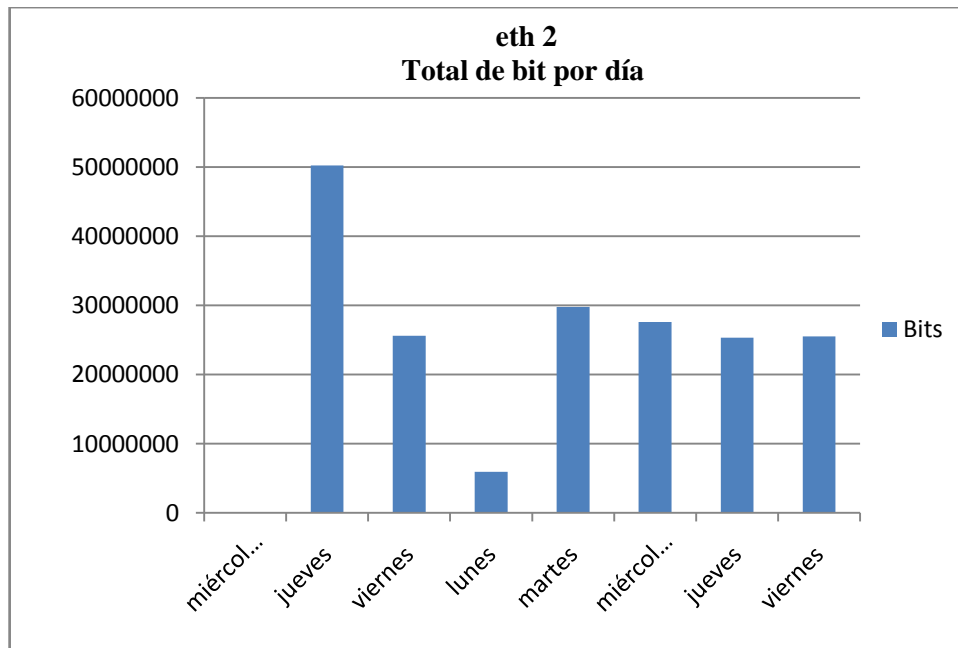
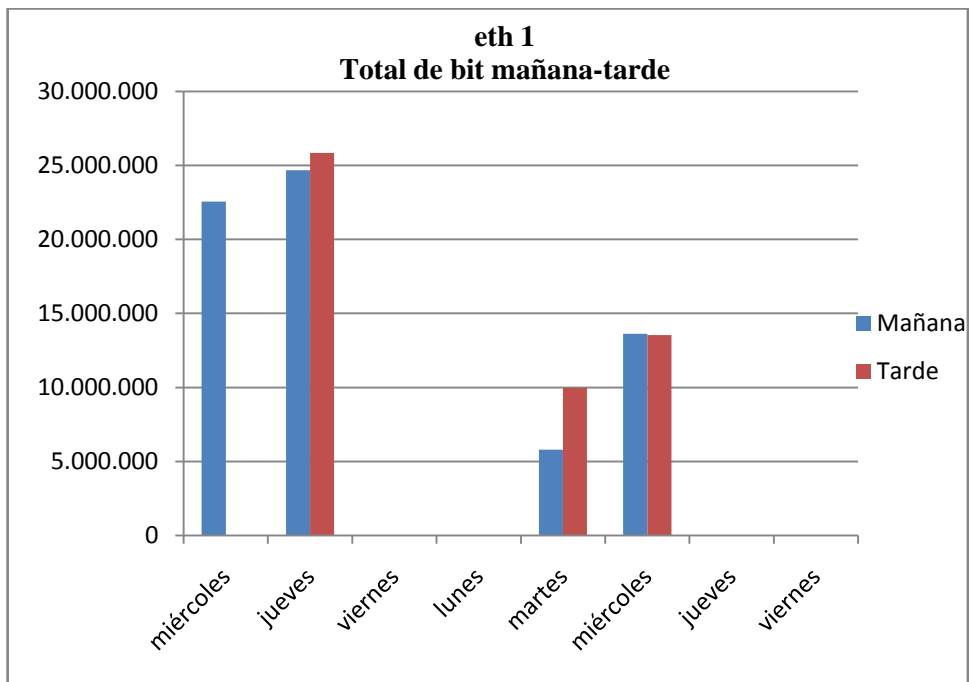
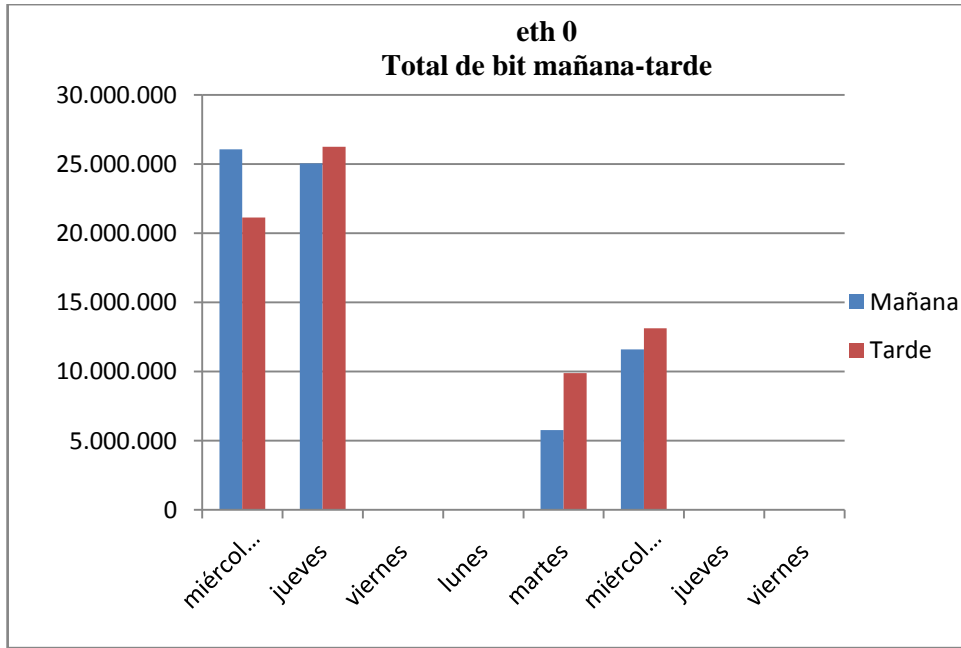


Figura. 5.21. Comportamiento de la cantidad de paquetes que circularon en la red por días, por cada estación.

Se puede decir que a partir del miércoles la circulación de paquetes va disminuyendo, así mismo el día lunes es el que tiende a una menor circulación que aumenta entre los días martes y miércoles, para luego disminuir.

Presentamos ahora el comportamiento según la mañana y la tarde:



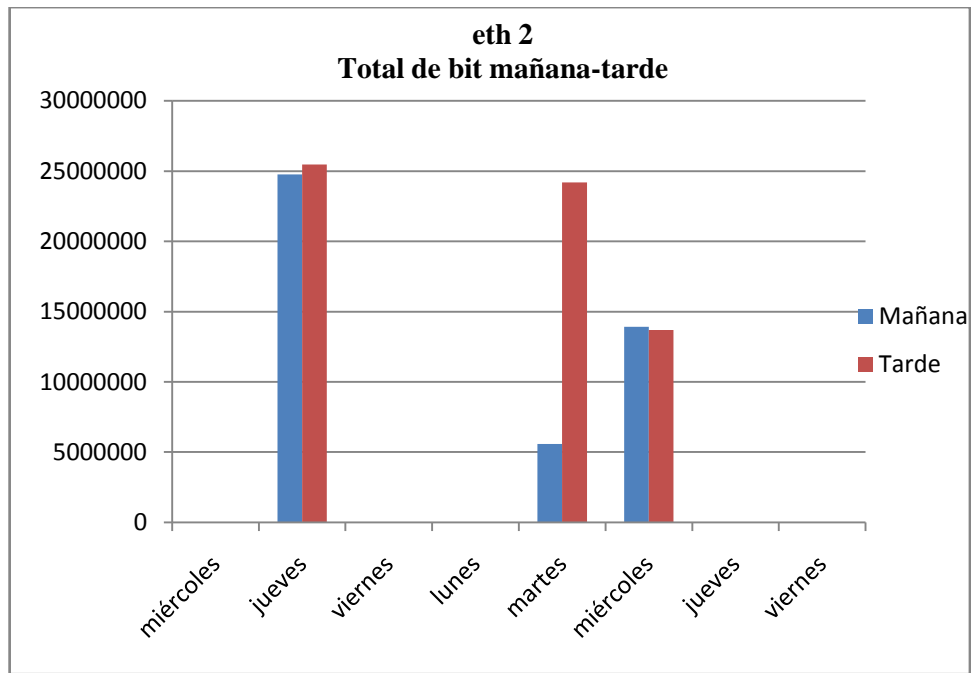


Figura. 5.22. Comportamiento de la cantidad de paquetes que circularon en la red por la mañana y la tarde

Aquí en sentido general la cantidad de paquetes en la tarde es mucho mayor que en la mañana como tendencia.

En sentido general el comportamiento de la circulación en la red es de la siguiente manera:

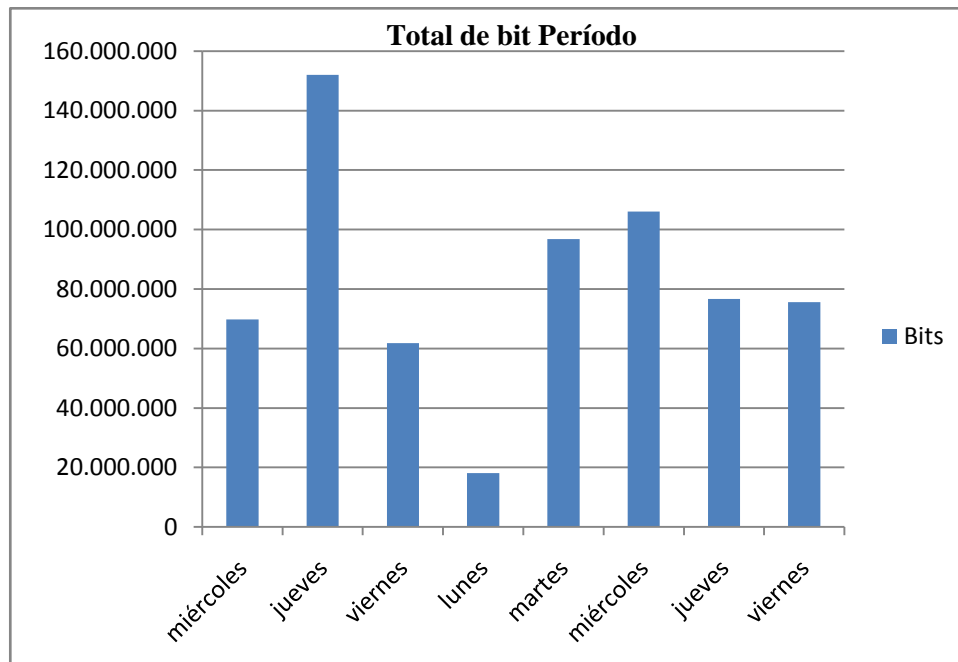


Figura. 5.23. Comportamiento de la Circulación en la red

Puede apreciarse la tendencia de mayor circulación los miércoles y jueves, y el lunes como el de menor circulación.

Finalmente se puede decir que la seguridad de una red no se basa en una única técnica exclusivamente, y particularmente con el modelo realizado del comportamiento normal del tráfico TCP/IP de la red, se tiene una herramienta adicional y una de las más efectivas en la detección de posibles intrusiones, que reforzada por la utilización de multitud de tecnologías que permiten monitorizar y gestionar cada uno de los aspectos críticos de la red: encriptación, IDSs, *firewalls*, software específico de seguridad, protocolos seguros (SSL, SSH, IPSec), PKIs) se pueden alcanzar altos niveles de seguridad.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

- En este proyecto de tesis se han revisado las bases más importantes en las cuales están implementados los protocolos TCP/IP, y las principales vulnerabilidades que estos presentan en su configuración, y de esta manera conocer el funcionamiento de estos protocolos dentro de un sistema de comunicación basado en redes de conmutación de paquetes.
- El protocolo TCP/IP sufre algunos problemas de seguridad por las características intrínsecas de su diseño, los cuales han sido analizados a lo largo de este capítulo. Tratando de permitir al administrador de red controlar y disolver muchas de las vulnerabilidades presentadas a lo largo del trabajo.
- El trabajo se ha enfocado en conocer como se manipulan las falencias de los protocolos TCP/IP, para violar la seguridad del sistema, ofreciendo información y mecanismos suficientes como para conseguir un nivel de seguridad aceptable en la red.
- Es por tanto necesario dedicar tiempo y esfuerzo a evolucionar la red hacia un entorno seguro, siendo necesario mantenerse actualizado de los avances que se realizan en este campo, así como de los nuevos avisos, vulnerabilidades y tecnologías que salen a la luz.
- Finalmente, cabe concluir con una reflexión al respecto de la seguridad: el esfuerzo dedicado a la protección de un entorno debe ser directamente proporcional al valor de su contenido. Debido a que toda vulnerabilidad posee su correspondiente protección, la vertiginosa carrera en la que la seguridad de las redes se debate actualmente, se centra en el

mantenimiento constante y actualizado de las protecciones necesarias para evitar las vulnerabilidades existentes ya conocidas.

- Mediante la utilización de herramientas de monitoreo y seguridad se puede lograr identificar las principales características que rigen el comportamiento normal del tráfico TCP/IP de cualquier red.
- Una vez finalizado y realizado el análisis del tráfico obtenido durante la semana, se puede obtener las principales características que rigen el comportamiento del tráfico TCP/IP de la red del DEEE.
- Mediante las diferentes simulaciones de diferentes ataques producidas a la red de pruebas, que buscan atacar principales vulnerabilidades del protocolo TCP/IP, se pretende dar a conocer las principales características de estos ataques con la finalidad de poder identificarlos cuando la situación lo amerite.
- Al finalizar cada una de las simulaciones de los diferentes ataques que explotan las vulnerabilidades del protocolo TCP/IP se puede concluir que con una simple petición que tenga éxito por parte de un atacante el sistema puede verse comprometido o a su vez ser utilizado como lanzadera contra otros ordenadores.
- Para solucionar el problema de la seguridad en una red, como ya se ha comentado a lo largo del proyecto, existen dos soluciones que se debería intentar aplicar: en primer lugar la concienciación de los problemas que pueden acarrear los fallos de seguridad. Para posteriormente, una formación adecuada, simplemente hay que informarse y conocer normas de seguridad y métodos empleados en la ejecución de un ataque y la manera de identificarlos.

-
- La seguridad de las redes de comunicaciones, evoluciona a pasos agigantados cada minuto que transcurre, nuevas vulnerabilidades y utilidades, tanto para explotarlas como para combatirlas, son distribuidas públicamente en la red. La información disponible esta al alcance, por lo que el diseño de un sistema de seguridad debe basarse en la fortaleza de las tecnologías existentes.
 - Aparte del lógico incremento en el nivel de seguridad que se puede conseguir mediante una concienciación y formación de los usuarios y administradores de red, existe un escollo que estas dos medidas difícilmente nos van a permitir superar: la simpatía que socialmente despiertan muchos piratas informáticos.
 - Este trabajo muestra un tipo de información que puede ser extraída de la carga de tráfico de una red, y que puede ser útil para los administradores de estas en la toma de decisiones en el diseño y desempeño de la red.
 - Además puede usarse los resultados obtenidos como referencia para crear modelos de simulación y llevar a cabo un análisis más detallado y predicciones sobre el comportamiento de la red para detección de anomalías y posibles ataques.
 - Para realizar un análisis de tráfico satisfactorio, es prioritaria la elección del modelo de tráfico que simule el comportamiento del tráfico del sistema deseado.
 - No todos los modelos de series de tiempo caracterizan apropiadamente el tráfico de una red, las series de tiempo como la de Promedios Móviles (MA) no logra capturar toda la dinámica de la serie; otros modelos como el AutoRegresivo (AR) proporcionan realizaciones bastante parecidas en magnitud a las originales, pero con un número grande de parámetros aumenta el costo computacional. Las series de tiempo y en especial los modelos AutoRegresivos e Integrados con Promedios Móviles (ARIMA),

resultan apropiados para modelar el tráfico en redes de datos Wi-Fi, así como otros modelos de tráfico se pueden ajustar con características de correlación al comportamiento del tráfico en diferentes tipos de redes, sin embargo mediante este trabajo lo que se busca es un modelo único que caracterice el tráfico TCP/IP de la red del DEEE.

- La idea de detectar intrusiones en la transferencia de paquetes de datos por medio de la estadística de predicción en series temporales, nos lleva al análisis exhaustivo en el desarrollo de una metodología para el control del tráfico que circula, al llegar nuevos datos se determina si pertenecen al intervalo de confianza y en caso contrario se lo evalúa determinando si se trata de una posible intrusión, la cual se la puede detectar en tiempo real analizando las características del tráfico.
- La importancia de la implementación de un modelo de tráfico, radica en la detección de nuevos ataques y en la evaluación constante del desempeño de la red. Un sistema de seguridad configurado bajo un modelo de operatividad normal de tráfico valida la cantidad de tráfico que circula en un instante determinado, comparando mediante un análisis estadístico predictivo si los parámetros y características que rigen al tráfico salen de sus umbrales normales de operación. Detectando anomalías que pueden ser indicio de la presencia de ataques o falsas alarmas.
- La extrema variabilidad de la carga de tráfico en cualquier tiempo o en cualquier intervalo de tiempo hace difícil especificar un patrón de tráfico en particular para caracterizar el tráfico de la red, Sin embargo puesto que los patrones de tráfico "Bursty", parecen ser típicos, los modelos del tráfico de redes se diseñan para acomodarse a esta propiedad.
- A parte de la identificación de que el tráfico de la red es "Bursty", otra caracterización es la cantidad de tráfico que circula en un tiempo determinado.

- Las evaluaciones realizadas al modelo mediante el método estadístico alrededor de los residuales, reflejan datos predictivos con un alto grado de desempeño, confiabilidad y exactitud en sus pronósticos.
- En líneas generales esta investigación contiene dos secciones fundamentales uno que da a conocer de manera objetiva la metodología del control de tráfico TCP/IP basada en un modelo de comportamiento normal que alerte cuando se produce una anomalía en la red, y el otro el uso de programas de monitorización y herramientas que permitan identificar si el tráfico que circula contiene características intrusivas que perjudiquen la integridad y seguridad de la red.
- Si bien la estimación de parámetros y modelos es aún un proceso complicado (en el sentido que no representan con precisión absoluta el tráfico real) y diariamente surgen nuevas características, es claro que sin un modelado adecuado no será posible monitorizar el desempeño de las redes, ni entender la naturaleza del tráfico, lo cual puede ser determinante a la hora de garantizar niveles de seguridad
- Adicionalmente la seguridad sobre las redes es un trabajo arduo, y de la misma forma que surgen soluciones, surgen también problemas, los cuales impactan significativamente la economía, por lo cual todas las aproximaciones en su búsqueda son válidas, lo que no se puede perder en el horizonte es que si las intrusiones son detectables desde el comportamiento de la red es posible detenerlas antes que afecten negativamente el desempeño de la red o los equipos conectados a ella.

RECOMENDACIONES

- Se recomienda incentivar este tipo de proyectos de investigación con la finalidad de que se invierta en ellos y se los pueda llegar a implementar.
- La seguridad en redes es una ciencia en constante evolución, que debe ser tratada con la importancia que se merece, por lo tanto continuamente se debería desarrollar nuevos métodos más eficientes en la detección de posibles ataques e intrusiones.
- Un Modelo del comportamiento normal de operación del tráfico que circula en una red, constituye un herramienta fundamental de monitoreo, sin embargo se recomienda una actualización permanente para obtener un mejor desempeño.

REFERENCIAS BIBLIOGRÁFICAS

1. GARCÍA ALFARO, Joaquín / Xavier Perramón Tornil (*Aspectos avanzados de Seguridad en Redes*)
2. VERDEJO ÁLVAREZ, Gabriel– “SEGURIDAD EN REDES IP: Los protocolos TCP/IP” .
3. Redes de Información (RIN) - Universidad Tecnológica Nacional – F.R.C
4. RAUL SILES PELAEZ. Análisis de seguridad de la familia de protocolos TCP/IP y sus servicios asociados, Edición I – Junio 2002.
5. PAT EYLER. Guía Avanzada Linux con TCP/IP, Pearson Educación, SA, Madrid 2001.
6. JORDI HERRERA JOANCOMARTI; JOAQUIN GARCÍA ALFARO; XAVIER PERRAMON TORNIL. *Aspectos Avanzados de seguridad en redes, 1ra Edición.*
7. JOSE LUIS RAYA CABRERA; CRISTINA RAYA PEREZ; Redes Locales Madrid 2002; ALFAOMEGA Grupo Editor
8. JEIMY J. CANO Ph.D. Análisis de tráfico de red Patrones normales Vs. Ataque; Universidad de los Andes.
9. FERNANDO GONT; Resultados de un análisis de seguridad de los protocolos TCP e IP; 4ta Jornada de Seguridad Informática; 25 de Noviembre de 2008, Paraná, Entre Ríos, Argentina
10. “BLACK HACK” (“El hacker Negro”); El Libro Negro del Hacker (Versión Digital)
11. BEJTLICH, Richard; *El TAO de la Monitorización de Seguridad en Redes*; Editorial PEARSON, 2005 España.
12. NORTH CUTT, Stephen; NOVAK, Judy, *Detección de Intrusos*, Editorial PEARSON EDUCACION, Segunda Edición, 2001 Madrid
13. LOS CUADERNOS DE HACK X CRACK; Escaneando la Red; Numero 13; Editotrans; Tarragona España
14. LOS CUADERNOS DE HACK X CRACK; TCP/IP; Numero 18; Editotrans; Tarragona España.

15. LOS CUADERNOS DE HACK X CRACK, No 20 (TCP- La esencia de las comunicaciones por Red).
16. LOS CUADERNOS DE HACK X CRACK, No 21 (TCP- Comprendiendo los ataques de Internet).
17. LOS CUADERNOS DE HACK X CRACK; La capa IP Encaminamiento de paquetes; Numero 25; Editotrans; Tarragona España.
18. LOS CUADERNOS DE HACK X CRACK; El corazón de Internet Protocolo TCP; Número 25; Editotrans; Tarragona España.

Bibliografía de Internet

1. <http://www.scribd.com/doc/2932879/Seguridad-de-Redes>
2. <http://seguridadyredes.nireblog.com/post/2008/01/17/analisis-capturas-trafico-red-interpretacion-datagrama-ip-parte-i>
3. <http://www.allbusiness.com/information/publishing-industries/722591-1.html>
4. http://catarina.udlap.mx/u_dl_a/tales/documentos/lep/mejia_s_ja/capitulo2.pdf
5. <http://www.inf.utfsm.cl/~rmonge/seguridad/cripto-07-bn.pdf>
6. <http://www ldc.usb.ve/~poc/Seguridad-viejo/tcpip.pdf>
7. http://www.wikilearning.com/tutorial/deteccion_de_intrusos-pasosaseguir_para_detectar_a_un_intruso/6428-2
8. <http://www.grupofact.cl/img/ar421.pdf>
9. <http://www.trucoswindows.net/conteni5id-48-SEGURIDAD-Resumen-de-los-tipos-de-ataques-mas-comunes.html>
10. <http://www.cpni.gov.uk/Docs/InternetProtocol.pdf>
11. <http://tcpreplay.synfin.net/wiki/WikiStart#WelcometoTcpreplay>
12. <http://nmap.org/download.html>
13. <http://www.springerlink.com/content/u1mgf81lj1ll5vra/?p=30569cafbf334abea491d35e804c6333&pi=10>
14. <http://www.icir.org/enterprise-tracing/>
15. <http://es.kioskea.net/contents/attaques/attaque-syn.php3>
16. http://foro.elhacker.net/hacking_basico/ip_spoofing_y_syn_flooding_con_necat_y_nemesis-t248361.0.html%3Bmsg1201249

17. http://foro.elhacker.net/seguridad/ataques_syn_flood-t247247.0.html
18. <http://www.dcc.uchile.cl/~hsalgado/dos/siframes.htm>
19. <http://el-directorio.org/Seguridad/Tools/Nmap#head-a297e1886183c1bb5b35d294d9d8973e4816ef28>
20. <http://www.angelfire.com/darkside/thc/espanol/root.html>

GLOSARIO

AR (Autoregressive)

ARP (Address Resolution Protocol)

ARIMA (Autoregressive Integrated Moving Average)

DARPA (Defense Advanced Research Projects Agency)

DEEE (Departamento de Eléctrica y Electrónica de la ESPE)

DF (Don't Fragment)

DLC (Data Link Control)

DNS (Domain Name System)

DOS (Deny of service)

Eth (Interface de red)

FTP (File Transfer Protocol)

HTTP (Hypertext Transfer Protocol)

IANA (Internet Assigned Numbers Authority)

ICMP (Internet Control Message Protocol)

IHL (Longitud de la cabecera)

IP (Internet Protocol)

LAN (Local Area Network)

MAC (Media Access Control Address)

MF (More Fragments)

MTU (Maxim Transfer Unit)

MCO (Mínimos Cuadrados Ordinarios)

NMAP (Network Mapper)

Nslookup (Name System Lookup)

OSI (Open Systems Interconnection)

SMTP (Simple Mail Transfer Protocol)

SSH (*Secure Shell*)

TCP (Transport Control Protocol)

TFTP (Trivial File Transfer Protocol)

TOS (Type of Service)

TTL (Time to Live)

TFN (Tribe Flood Network)

TFNK2k (Tribe Flood Network 2000)

UDP (User Datagram Protocol)

ANEXOS

A continuación se adjunta un ejemplo del tráfico capturado en un día correspondiente a la interfaz eth0.

ESCUELA POLITÉCNICA DEL EJÉRCITO

**CARRERA DE INGENIERIA ELECTRÓNICA EN
TELECOMUNICACIONES**

Fecha de entrega.....

Mario Fernando García Guerra

Ing. Gonzalo Olmedo, Ph.D
COORDINADOR DE CARRERA