



CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS
AVANZADOS DEL INSTITUTO POLITÉCNICO
NACIONAL

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA
SECCIÓN DE COMPUTACIÓN

Sistema de Supervisión Automática de Redes Locales

Tesis que presenta

Isaí Cortés Mojica

Para obtener el grado de
Maestro en Ciencias
en la Especialidad de
Ingeniería Eléctrica
opción Computación

Director de la tesis:

Dr. Arturo Díaz Pérez

México, D.F.

Julio 2005

ABSTRACT

Local area networks (LANs) have increased its use because they offer great benefits: sharing of resources, high speed of transmission, etc. Despite many hardware and software tools have been created to guarantee the availability, integrity and confidentiality of the information that circulates through LANs, these tools do not automatically analyze the network traffic to detect anomalous behavior (i.e. failures, attacks or intruders). This arduous task is made manually by network administrator.

This thesis presents a system that automatically detects anomalous behaviors in LANs. Our system uses a set of dynamic statistical models. These models are automatically updated with respect to the network daily behavior. Furthermore, as an additional feature, it gives the capacity to measure the network performance at a device level detail (hubs, switches, gateways and computers).

Our system was developed using Perl and Tk. It analyzes the network behavior and the network topology to automatically detect anomalous behavior. It graphically show the affected route and the broken device. Data of the behavior of packages, ports, physical location, etc. are available to the administrator via a graphical Web interface.

RESUMEN

Las redes de área local han ganado gran aceptación gracias a que ofrecen grandes beneficios como son: disminución de costos, la compartición de recursos entre usuarios, alta velocidad de transmisión, entre otros. Las redes locales deben trabajar óptimamente para garantizar la disponibilidad, integridad y confidencialidad de la información que circula por las mismas. Con este propósito, se han creado herramientas de hardware y software que monitorean el tráfico de una red de área local. Sin embargo, estas herramientas no hacen un análisis de los datos recuperados, tarea que es dejada al administrador de la red. Esta es una labor ardua, ya que el administrador debe analizar los datos de la red en busca de fallas o comportamientos anormales como ataques o intrusiones de usuarios no autorizados.

Esta tesis presenta un sistema capaz de detectar automáticamente comportamientos anómalos en una red de área local, ocasionados ya sea por fallas o ataques a la misma. Nuestro sistema utiliza un conjunto de modelos estadísticos dinámicos que es actualizado automáticamente con el comportamiento diario de la red. Además, incorpora un editor de topologías de red, con el cual se puede medir el comportamiento normal o anormal de cada dispositivo de red, incluyendo concentradores, switches, puertos de enlace y computadoras personales.

Nuestro sistema fue desarrollado en Perl y Tk. En base al comportamiento y la topología de la red (tomados de una base de datos), nuestro sistema es capaz de detectar automáticamente comportamientos anómalos, con la capacidad de localizar gráficamente la ruta afectada y rastrear el dispositivo fallido. Datos sobre el comportamiento de paquetes, puertos utilizados, ubicación física, etc. son puestos a disposición del administrador mediante una interfaz Web gráfica.

A mi mamá **Lupita** y papá **Adrián** de quienes me siento orgulloso
y han hecho de mi vida un camino lleno de amor, apoyo, comprensión
y consejos, pero sobre todo feliz.

A mis hermanos **Adriana**, **Abraham**, **Daniel** y **Raquel** por su apoyo,
fé y hacer divertidos todos los momentos que pasamos juntos.

A **Jessica** por traer alegría extra a mi vida.

Agradecimientos

Gracias a mis abuelitos por su amor, cariño y hacer de mi infancia una etapa grata de recordar.

Gracias a mis tíos por su calidez y apoyo.

Gracias a mis primos por su amistad y por que cada momento con ellos es digno y divertido de recordar.

Gracias a mis padres y hermanos por su comprensión y apoyo incondicional.

Gracias a mis amigos quienes hicieron de mi estancia en el CINVESTAV un rato agradable y ameno.

Agradezco a mi asesor Dr. Arturo Díaz Pérez por su paciencia en la realización de esta tesis, por sus numerosos consejos y sobre todo por transmitirme valiosos conocimientos.

Agradezco a mis sinodales, los Dres. Luis Gerardo de la Fraga y Guillermo Morales Luna por su colaboración en beneficio de la calidad de esta tesis.

Gracias a las secretarias de la Sección de Computación, en especial a la Sra. Sofía Reza por su dedicación y gusto al realizar su trabajo, haciendo de los tramites un proceso sencillo y ameno.

Agradezco a la Sección de Computación por facilitarme las instalaciones para el desarrollo de mis estudios de maestría.

Agradezco al CINVESTAV por su apoyo al permitirme cursar todos los cursos necesarios para el desarrollo de este trabajo de tesis.

Agradezco al CONACyT por la beca proporcionada durante mi estancia en el programa de maestría.

Índice general

1. Introducción	1
2. Seguridad en redes de cómputo	5
2.1. Redes locales	5
2.1.1. Topología de red	8
2.2. Seguridad	9
2.3. Seguridad en redes	10
2.4. Sistemas de detección de intrusos	13
2.4.1. Clasificación general	14
2.4.2. Fuentes de información	14
2.4.3. Tipos de análisis	16
2.4.4. Tipos de respuesta a incidentes	20
2.4.4.1. Respuestas pasivas	21
2.4.4.2. Falsas alarmas	22
2.5. Medición del tráfico	22
2.6. Modelado de tráfico	23
3. Supervisión automática de redes locales	25
3.1. Sistema monitor de tráfico de redes locales	26
3.2. Análisis automático del tráfico de red	28
3.2.1. Modelo estadístico dinámico	29
3.2.2. Topología de una red de área local	34
3.3. Detección automática de comportamientos anómalos	45
3.3.1. Detección basada en anomalías	47
3.3.2. Comportamientos anómalos basados en reglas	51
3.3.2.1. Exploración de puertos	52

3.3.2.2. Exploración de direcciones IP	53
4. Supervisor de red	55
4.1. Base de datos	56
4.2. Detección de fallas y abusos	60
4.3. Rastreo de dispositivos anómalos	61
4.4. Actualización de modelos estadísticos	62
4.5. Detección de intentos de intrusión	62
4.6. Simulador de tráfico de una red de área local	64
4.6.1. Alteración del modelo estadístico dinámico	66
4.6.2. Generación de tráfico de red artificial	67
4.6.3. Generación de fallas y abusos artificiales	68
4.6.4. Generación de exploraciones de red artificiales	68
5. Interfaz de usuario	73
5.1. Interfaz Web	75
5.1.1. Tráfico de entrada y salida	75
5.1.2. Detalles de tráfico en la red local	77
5.1.3. Tráfico por dispositivo de interconexión	79
5.1.4. Direcciones IP por dispositivo de interconexión	81
5.2. Editor de topología TKETOP	82
5.3. Reportes de comportamientos anómalos	86
6. Conclusiones y trabajo futuro	91
Bibliografía	95

Índice de figuras

2.1. Número de incidentes por meses	12
2.2. Incidentes por tipo	13
2.3. Tipos principales de sistemas detectores de intrusos	15
3.1. Diagrama del sistema monitor de tráfico de redes locales.	26
3.2. Modelo estadístico del comportamiento de una red de área local	31
3.3. Modelo estadístico dinámico comparado con el comportamiento real de la red	33
3.4. Gráfica de error del modelo estadístico dinámico	35
3.5. Partes de un árbol	36
3.6. Tráfico de red en una topología de árbol	37
3.7. Escenarios de un comportamiento anómalo en dispositivos de interconexión.	40
3.8. Red de área local.	41
3.9. Modelo estadístico dinámico de HUB1	42
3.10. Modelo estadístico dinámico de HUB2	42
3.11. Modelo estadístico dinámico de SWITCH	43
3.12. Tráfico de red durante una falla	47
3.13. Tráfico de red durante un abuso	48
3.14. Tráfico de red durante un ataque	50
3.15. Tráfico de red durante la proliferación de un gusano	51
4.1. Arquitectura general del sistema de supervisión de redes de área local	56
4.2. Tablas de la base de datos.	57
4.3. Arquitectura del simulador de tráfico de red	66
5.1. Estructura del sistema Web	74

5.2. Gráfica del comportamiento del tráfico de entrada mostrada en la interfaz Web	75
5.3. Gráfica de error entre el tráfico de entrada real y el tráfico de entrada esperado por el modelo estadístico	76
5.4. Tabla de detalles de comportamiento general	77
5.5. Datos específicos por dispositivo de red	78
5.6. Selección de dispositivos de interconexión	79
5.7. Gráfica de comportamiento de un dispositivo de interconexión	80
5.8. Gráfica de comportamiento de un dispositivo de interconexión por protocolos	81
5.9. Tabla de detalles de comportamiento por dispositivo de interconexión .	82
5.10. Menú principal de TkETOP	82
5.11. Ventana Insertar	83
5.12. Ventana Modificar	84
5.13. Ventana Eliminar	84
5.14. Ventana Búsqueda	85
5.15. Mapa de la topología generado por TkETOP	86
5.16. Red de área local simulada.	87
5.17. Gráfica de comportamiento de SWITCH1	87
5.18. Reporte de fallas y abusos	88
5.19. Reporte de exploración de puertos	89

Índice de cuadros

3.1.	Nivel de tolerancia para comportamientos anómalos	34
3.2.	Evolución del valor $\Delta e(d, h, c)$ del comportamiento mostrado dentro del rectángulo en la Figura 3.4.	35
3.3.	Pares padre-hijo de la Figura 3.6	38
3.4.	Evolución del valor $\Delta e(d, h, c)$ y rastreo de dispositivos anómalos de la red mostrada en la Figura 3.8	44
3.5.	Características de comportamientos anómalos	50
3.6.	Estructura de la tabla hash usada para la exploración de puertos	52
3.7.	Comportamiento de los pares IP+puerto y Δp durante una exploración de puertos	52
3.8.	Estructura de la tabla hash usada para la exploración de direcciones IP	53
3.9.	Comportamiento de los pares IP+puerto y Δd durante una exploración de direcciones IP	53
4.1.	Campos correspondientes a la tabla <i>datos</i>	57
4.2.	Campos correspondientes a la tabla <i>tot_par</i>	58
4.3.	Campos correspondientes a la tabla <i>gra24</i>	58
4.4.	Campos correspondientes a la tabla <i>des_maq</i>	59
4.5.	Claves numéricas correspondientes al campo <i>tipo</i> perteneciente a la tabla <i>des_maq</i>	59
4.6.	Campos correspondientes a la tabla <i>top_maq</i>	59
4.7.	Campos correspondientes a la tabla <i>com_disp</i>	60
4.8.	Claves numéricas correspondientes al campo <i>num_dia</i> de la tabla <i>com_disp</i>	60
4.9.	Porcentaje del uso de protocolos	68

Índice de algoritmos

1.	Detección de fallas y abusos en dispositivos de interconexión	61
2.	Rastreo automático de dispositivos anómalos fuente	62
3.	Retroalimentación de los modelos estadísticos dinámicos	63
4.	Detección de exploraciones de red	64
5.	Alteración del modelo estadístico	67
6.	Generación de tráfico de red artificial	69
7.	Generación de fallas y abusos	69
8.	Generación de exploraciones de red	71

Capítulo 1

Introducción

Los mecanismos de seguridad existentes protegen a las computadoras y a las redes de ser usadas por personas no autorizadas mediante claves y controles de acceso. Sin embargo, si éstos son comprometidos o violados, un intruso puede ganar acceso no autorizado y de esta manera causar daño o comprometer la integridad del funcionamiento de la red. Por otro lado, el desempeño de la red puede verse perjudicado a causa de fallas o mal funcionamiento presentado por los distintos dispositivos de interconexión que conforman una red de área local.

A pesar de que la primera defensa de una computadora son sus controles de acceso, es claro que no es posible confiar tan sólo en estos mecanismos para proteger el sistema contra intrusiones, ataques internos, virus o ladrones de información.

Ya sea una falla, abuso o intrusos, son propensos a presentar un patrón de comportamiento el cual difiere notablemente del presentado por un comportamiento normal. Estos comportamientos anómalos pueden ser detectados a través de la observación estadística de su comportamiento atípico. Sin embargo, es necesario el desarrollo de herramientas capaces de analizar de manera automática la información correspondiente al comportamiento presentado por la red, ya que debido al gran volumen de información que éste genera, resulta una tarea ardua para el administrador de red. Esta idea es la base del desarrollo de sistemas basados en la observación automática del comportamiento de los sistemas para llevar a cabo la detección automática de comportamientos atípicos.

Existen dos métodos principales para la detección de comportamientos atípicos, éstos son los métodos *basados en reglas* [30, 42] y *basados en anomalías* [8, 15]. Los primeros observan el comportamiento de la red en busca de comportamientos anómalos

previamente conocidos y almacenados en una base de conocimiento. Por otro lado los métodos basados en anomalías buscan comportamientos que difieran de un comportamiento típico presentado por el sistema.

Cada método tiene sus propias ventajas y desventajas. Los métodos basados en reglas cuentan con la ventaja de presentar un rango de falsas alarmas bastante bajo. Sin embargo, no tienen la capacidad de detectar intrusiones o abusos para los cuales no tienen un conocimiento previo. Por otro lado, los métodos basados en anomalías son capaces de detectar cualquier tipo de comportamiento atípico aunque no cuenten con un conocimiento previo del mismo. La desventaja de utilizar este mecanismo reside en la alta frecuencia que puede llegar a tener de falsas alarmas.

Tratar de predecir el comportamiento del tráfico circulante de una red de área local resulta un problema complicado. Debido a la gran cantidad de factores involucrados en el proceso. Los principales factores involucrados son el tipo de tráfico circulante (entrada o salida), día y hora analizados (hábil o no hábil).

Existen en la actualidad una gran variedad de estudios, los cuales proponen diversos métodos para la predicción del comportamiento del tráfico de red [22, 10, 18]. La mayoría de estos métodos presentan un balance entre eficiencia y precisión. Mientras que unos métodos son muy rápidos resultan ser poco precisos. Otros son muy exactos en sus resultados pero demasiado lentos para cumplir con los propósitos de seguridad.

Un modelo estadístico es un método que mantiene un balance entre rapidez y precisión, permitiendo determinar de manera aproximada el valor de una respuesta observada con base en un análisis estadístico previo al evento en cuestión [22]. Un modelo estadístico del comportamiento del tráfico de la red es capaz de calcular un comportamiento futuro con base en la observación del comportamiento del tráfico de red presentado previamente.

Los modelos estadísticos se pueden clasificar en *estáticos* o *dinámicos*. Un modelo estadístico estático se obtiene a partir de una gran cantidad de datos obtenidos previamente. Un modelo estadístico dinámico es capaz de actualizarse de manera automática observando y analizando los datos obtenidos continuamente.

Se propone un método capaz de detectar fallas y abusos presentados en la red de área local. Dicho método consiste en utilizar un modelo estadístico dinámico por cada dispositivo de interconexión que conforma la red de área local. Cada modelo tiene como objetivo servir como punto de comparación del comportamiento de tráfico presentado por los dispositivos de interconexión. Entonces una toma de decisiones puede ser efectuada.

Con el propósito de realizar un análisis más detallado de lo que sucede en una red local, es necesario contar con la información topológica de la red de área local. Con dicha información, además de proporcionar los datos necesarios para generar los modelos, se proporciona un nivel más alto de abstracción de la red de área local, lo cual se traduce en un mayor nivel de precisión al momento de detectar comportamientos anómalos.

El propósito del presente trabajo es desarrollar un sistema que sea capaz de detectar de manera automática comportamientos anómalos en una red de área local. Para ello se propone el uso de modelos estadísticos dinámicos por cada dispositivo de interconexión dentro de la red. Contar con un modelo estadístico dinámico por cada dispositivo de interconexión, permite al sistema tener la capacidad de detectar automáticamente de manera precisa comportamientos anómalos en los distintos nodos de la red de área local. Aunado a esto, se complementa la capacidad de detección del sistema mediante la incorporación de un método capaz de detectar intentos de intrusión tales como exploraciones de puertos y direcciones IP.

En la tesis "Análisis del Tráfico de una Red Local", desarrollada por Jorge Enrique Morfín Galván [10], se describe un sistema monitor de red, el cual permite capturar todos los paquetes entrantes y salientes que circulan a través de una red de área local, almacenar la información del encabezado de cada uno de ellos en una base de datos, así como realizar un análisis estadístico. Sin embargo, este sistema no realiza un análisis automático del tráfico de red en busca de comportamientos anómalos, dejando esta tarea al administrador de red. Por lo tanto se propone continuar el desarrollo del sistema integrando un conjunto de extensiones al mismo, logrando con ello un análisis automático del tráfico de red. Dichas extensiones consisten en actualizar el modelo estadístico estático a varios modelos estadísticos dinámicos correspondientes a los siete días de la semana, integrar la información de la topología de la red, obtener modelos estadísticos dinámicos de cada dispositivo de interconexión, así como integrar detectores que permitan utilizar los modelos dinámicos para la detección automática de algunos comportamientos anómalos.

Las extensiones se dividen en cuatro etapas. La primera es la etapa de análisis, la cual es la encargada de procesar la información obtenida del sistema monitor de red y distribuirla de acuerdo a la topología de la red. La segunda etapa es la encargada de comparar el comportamiento presentado por cada dispositivo de interconexión con su modelo estadístico dinámico. Esta comparación es realizada con la finalidad de detectar comportamientos anómalos. La tercera etapa es la encargada de retroalimentar los modelos estadísticos dinámicos con la información del comportamiento actual. La

cuarta etapa es la encargada de buscar comportamientos correspondientes a intentos de intrusión a la red.

El documento de tesis se encuentra organizado de la siguiente manera: en el Capítulo 2 se explican los tipos de redes de área local, las diferentes maneras en las que pueden estar conectadas así como los diferentes dispositivos que las componen. Además de plantear las principales amenazas de su seguridad y las diferentes maneras de protegerlas. Una explicación de los diferentes métodos existentes para la modelado de tráfico es vista también en este capítulo. En el Capítulo 3 se explica el funcionamiento del sistema monitor de red y la obtención de un modelo estadístico estático. Se indica cómo están constituidas las diferentes etapas del sistema desarrollado. En el Capítulo 4 se explica la arquitectura y el funcionamiento de cada módulo del sistema. En el Capítulo 5 se muestran los resultados obtenidos y la interfaz Web desarrollada. Finalmente, se establecen las conclusiones del trabajo desarrollado.

Capítulo 2

Seguridad en redes de cómputo

En un principio el número de computadoras que conformaban las redes locales era muy pequeño, la seguridad necesaria era mínima. Las redes se componían en su mayoría por una pequeña comunidad cuyos miembros eran de confianza. La mayoría de los datos que se intercambiaban no eran confidenciales. Por el contrario, las redes locales actuales requieren un mayor nivel de seguridad, manejan grandes volúmenes de información, atienden de forma independiente operaciones de distintos grupos que en muchas ocasiones intercambian datos privados. Es común la existencia de redes que diariamente reciben algún tipo de ataque con diversos fines, desde detener sus servicios hasta obtener algún tipo de dato confidencial.

En este marco, las necesidades de seguridad en redes de cómputo son importantes. El análisis del tráfico que circula por una red de área local, es la herramienta más básica para la detección de abusos y violaciones de seguridad a las que puede verse afectada una red de este tipo. Antes de explicar cómo funcionan las herramientas utilizadas para que una red sea segura, es necesario definir los diferentes tipos y características de las redes locales actualmente existentes, así como lo que se entiende por términos como seguridad, confianza y vulnerabilidad.

2.1. Redes locales

A finales de la década de los setenta, la Organización Internacional para la Normalización (ISO) empezó a desarrollar un modelo conceptual para la conexión en red al que bautizó con el nombre de *Open System Interconnection Reference Model*, mejor conocido como el modelo OSI [24] (pp. 8-11). El modelo OSI divide en siete capas el

proceso de transmisión de la información entre equipos informáticos, donde cada capa se encarga de ejecutar una determinada parte del proceso global. La *capa física* abarca los aspectos físicos de la red es decir, los dispositivos que conforman el entorno físico de la red. La *capa de enlace de datos* se encarga de establecer la manera en que las computadoras envían y reciben los datos a través del soporte físico proporcionado en la capa anterior. La *capa de red* encamina los datos determinando la ruta que deben seguir, además de efectuar el intercambio efectivo de los mismos dentro de dicha ruta. La *capa de transporte* es la encargada de controlar el flujo de datos entre nodos que establecen una comunicación; los datos no sólo deben entregarse sin errores, sino además en la secuencia adecuada. La *capa de sesión* es la encargada de establecer el enlace de comunicación entre las computadoras emisora y receptora. La *capa de presentación* toma los paquetes de la capa de aplicación y los convierte a un formato genérico que pueden leer todas las computadoras. La *capa de aplicación* proporciona la interfaz y servicios que soportan las aplicaciones de usuario.

Aunque existen diversos tipos de tecnologías para interconectar redes locales como por ejemplo, Token Ring [1] (pp. 31-33), FDDI [1] (pp. 326-327) y LocalTalk [24] (p. 132) en la actualidad la mayoría de ellas utilizan Ethernet a diferentes velocidades (10, 100 y 1000 Mbps) [24] (pp. 132-136). En 1985, las compañías DEC, Intel y Xerox formaron el subcomité IEEE 802.3, que estableció el estándar del mismo nombre bajo el cual se rige la red de área local conocida como Ethernet. Esta red usa el método de acceso CSMA/CD (*Carrier Sense Multiple Access / Collision Detection*), en el cual las computadoras compiten por el uso del medio de comunicación. En este método, cuando una computadora desea transmitir un paquete de datos, verifica el canal para ver si hay una señal portadora, si la hay, significa que el canal está siendo ocupado y la computadora espera un tiempo aleatorio antes de verificarlo otra vez. Si la computadora no detecta una señal, eso significa que el canal está libre y procede a enviar su paquete. Utilizando la norma IEEE, los equipos y protocolos de red pueden interoperar eficazmente.

Los protocolos de red son normas que permiten a las computadoras comunicarse, definen la forma en que las mismas deben identificarse entre sí en una red, la forma en que los datos deben transitar por la red, y cómo esta información debe procesarse una vez que alcanza su destino final. Los protocolos más usados son: IPX (para Novell NetWare) [24] (pp. 90-92), TCP/IP (para UNIX, Windows y otras plataformas) [1], DECnet (para conectar una red de ordenadores Digital) [7] (pp. 195-197), AppleTalk (para los ordenadores Macintosh) [24] (pp. 261-312), y NetBIOS/NetBEUI (para redes

LAN Manager y WindowsNT) [24] (pp. 187-194).

La familia de protocolos TCP/IP son los más usados por las redes actuales debido a su compatibilidad con distintos tipos de sistemas operativos y hardware, cubriendo distintos niveles del modelo OSI, siendo los dos protocolos más importantes el TCP (*Transmission Control Protocol*) y el IP (*Internet Protocol*). La arquitectura de TCP/IP consta de cinco niveles o capas en las que se agrupan los protocolos, y que se relacionan con los niveles OSI de la siguiente manera: *Aplicación*: Corresponde a los niveles OSI de aplicación, presentación y sesión. Aquí se incluyen protocolos destinados a proporcionar servicios, tales como correo electrónico (SMTP), transferencia de archivos (FTP), conexión remota (TELNET), protocolo HTTP (*Hypertext Transfer Protocol*), entre otros. *Transporte*: Coincide con el nivel de transporte del modelo OSI. Los protocolos de este nivel, tales como TCP y UDP se encargan de manejar los datos y proporcionar la fiabilidad necesaria en el transporte de los mismos. El protocolo UDP (*User Datagram Protocol*) proporciona una comunicación muy sencilla entre las aplicaciones de dos computadoras incorporando los puertos origen y destino en su formato de mensaje. El protocolo TCP (*Transmission Control Protocol*) permite una comunicación fiable entre dos aplicaciones. *Internet*: Es el nivel de red del modelo OSI. Incluye al protocolo IP, que se encarga de enviar los paquetes de información a sus destinos correspondientes. *Físico*: Análogo al nivel físico de OSI. *Red*: Es la interfaz de la red real, correspondiente a las interfaces 802.2, CSMA/CD, X.25, etc.

Para identificar globalmente una computadora dentro de un conjunto de redes TCP/IP se utilizan las direcciones IP (capa de red) [1] (pp. 54-56). Las direcciones IP se clasifican en: *Direcciones IP públicas*: Una computadora con una IP pública es accesible (visible) desde cualquier otra computadora conectada a Internet. *Direcciones IP privadas o reservadas*: Son visibles únicamente por otros huéspedes de su propia red o de otras redes privadas interconectadas por ruteadores. Las direcciones IP están formadas por 4 bytes (32 bits), los cuales se representan de la forma a.b.c.d, donde cada una de estas letras es un número comprendido entre el 0 y el 255. Por ejemplo: 129.42.18.99.

Una computadora puede estar conectada con distintos servidores a la vez. Para distinguir las distintas conexiones dentro de una misma computadora se utilizan los puertos [1] (pp. 12-13). Un puerto es un número de 16 bits que las aplicaciones utilizan para recibir y transmitir mensajes. Los números de puerto de las aplicaciones cliente son asignados dinámicamente y generalmente son superiores al 1024. En cambio, las aplicaciones servidoras utilizan unos números de puerto prefijados (*well-known*).

El objetivo de la interconexión de redes (*internetworking*) es dar un servicio de transporte de datos que involucre diversas redes con diferentes tecnologías de forma transparente para el usuario. Los dispositivos de interconexión de redes sirven para superar las limitaciones físicas de los elementos básicos de una red. Algunas de las ventajas que plantea la interconexión de redes de datos, son: Compartición de recursos dispersos, coordinación de tareas de diversos grupos de trabajo, reducción de costos al utilizar recursos de otras redes y aumento de la cobertura geográfica. Para lograr este objetivo se dispone de algunos dispositivos de interconexión de redes tales como: *Hub o Concentrador*: Este dispositivo repite simultáneamente la señal a múltiples cables conectados en cada uno de los puertos del hub. En el otro extremo de cada cable está un nodo de la red, por ejemplo una computadora personal. *Switch o Conmutador*: Los conmutadores ocupan el mismo lugar en la red que los concentradores, pero a diferencia de los concentradores, los conmutadores examinan cada paquete y lo procesan en consecuencia en lugar de simplemente repetir la señal a todos los puertos. *Gateway o Puerta de Enlace*: Es un equipo para interconectar redes con protocolos y arquitecturas completamente diferentes, a todos los niveles de comunicación. La traducción de las unidades de información reduce mucho la velocidad de transmisión a través de estos equipos. *Router o Ruteador*: Estos dispositivos envían paquetes de datos de un protocolo común, desde una red a otra, convierten los paquetes de información de la red de área local en paquetes capaces de ser enviados mediante redes de área extensa. Durante el envío, el ruteador examina el paquete buscando la dirección de destino y consultando su propia tabla de direcciones, la cual mantiene actualizada intercambiando direcciones con los demás ruteadores para establecer las rutas de enlace más adecuadas a través de las redes que los interconectan.

2.1.1. Topología de red

La topología se refiere a la forma en que están interconectados los distintos dispositivos de una red, tales como una computadora o un dispositivo de interconexión [12]. Una red tiene dos diferentes topologías: una física y una lógica. La topología física es la disposición física actual de la red, la manera en que los nodos están conectados unos con otros. La topología lógica es el método que se usa para comunicarse lógicamente con los demás nodos, la ruta que toman los datos de la red entre los diferentes nodos de la misma. Las topologías física y lógica pueden ser iguales o diferentes. Las topologías de red más comunes son: bus, anillo y estrella.

La topología de tipo *anillo* es en la cual los nodos están enlazados formando un círculo a través de un mismo cable. Las señales circulan en un solo sentido por el círculo, regenerándose en cada nodo. En la práctica, la mayoría de las topologías lógicas en anillo son en realidad una topología física en estrella. El anillo, como su propio nombre lo indica, consiste en conectar linealmente entre sí todos los nodos, en un ciclo cerrado. La topología de tipo *bus* consiste en que los nodos se unen en serie con cada nodo conectado a un cable largo o bus, formando un único segmento. Una rotura en cualquier parte del cable causará, normalmente, que el segmento entero pase a ser inoperable hasta que la rotura sea reparada. La topología de *estrella* se caracteriza por existir en ella un nodo central, al cual se conectan todos los equipos, de un modo muy similar a los radios de una rueda. De esta disposición se deduce el inconveniente de esta topología, y es que la máxima vulnerabilidad se encuentra precisamente en el nodo central, ya que si este falla, toda la red fallaría. A la interconexión de varias subredes en estrella se le conoce con el nombre de topología en *árbol o jerárquica*.

2.2. Seguridad

Según el enfoque formal, la seguridad se define a través de tres conceptos los cuales son [31]: *Confidencialidad* la cual, implica que la información sea accedida exclusivamente por el personal autorizado a la misma. La *integridad* consiste en la necesidad de tener la información inalterada. Y por último la *disponibilidad* se refiere a la necesidad de ofrecer un servicio ininterrumpidamente, de forma que pueda ser accedido en el momento en que se requiere, evitando en lo posible que algún tipo de incidencia detenga al mismo. Algunos estudios [21, 26] integran la seguridad dentro de una propiedad más general de los sistemas, la *confiabilidad*, entendida como el nivel nivel de calidad del servicio ofrecido.

A continuación se describen los términos de seguridad de redes más importantes. *Confianza*: La confianza es la esperanza que se tiene de que un sistema se comporte como realmente debería. Establecer relaciones de confianza sin garantías conlleva la aparición de vulnerabilidades, que se convierten en amenazas potenciales. *Vulnerabilidad*: Las vulnerabilidades son deficiencias o agujeros de seguridad del sistema que pueden ser utilizadas para violar las políticas de seguridad. Existen muchos tipos de vulnerabilidades. Pueden ser debidas a problemas en el diseño de una aplicación, ya sea de software o hardware. O también pueden ser debidas a un plan poco exhausti-

vo o insuficiente de políticas de sistema. *Amenaza*: Las amenazas son el resultado de explotar las vulnerabilidades. Una amenaza es una situación que tiene la capacidad de perjudicar o dañar el sistema. Aunque tanto las amenazas como las vulnerabilidades estén relacionadas, no son lo mismo. Un sistema de supervisión de red se debe encargar de identificar y responder a ambas.

2.3. Seguridad en redes

La seguridad en redes es el esfuerzo de crear una plataforma computacional segura, diseñada de tal manera que agentes (usuarios o programas) no sean capaces de efectuar acciones para las cuales no tienen permiso, pero puedan realizar las acciones que les son permitidas. Dichas acciones pueden ser operaciones de acceso, modificación y eliminación [7] (pp. 90-98).

Existen dos tipos de ataques que afectan a los sistemas y redes [11]: *Ataques activos*: Estos ataques implican algún tipo de modificación de los datos o la creación de falsos datos Suplantación de identidad, Modificación de mensajes, Web Spoofing Etc. *Ataques pasivos*: En los ataques pasivos el atacante no altera la comunicación, sino que únicamente la escucha o monitoriza, para obtener de esta manera la información que está siendo transmitida.

Exploraciones

Un ataque de exploración [3] es un tipo de ataque pasivo el cual, ocurre cuando un atacante examina la red o el sistema enviando diferentes tipos de paquetes. Usando las diferentes respuestas que envía el sistema afectado, el atacante puede descubrir muchas de las características y vulnerabilidades del sistema. Un ataque de exploración actúa como un identificador del sistema para el atacante. Los ataques de exploración no penetran el sistema o red afectada. Existen varios tipos de herramientas usadas para realizar este tipo de ataques, entre las más conocidas se incluyen: *exploraciones de direcciones IP*, *exploraciones de puertos* y *exploraciones de vulnerabilidades*. Los ataques de exploración pueden descubrir: La topología de la red, los tipos de tráfico de red permitidos por un cortafuegos, los huéspedes activos de la red, el sistema operativo utilizado por los huéspedes, el software de servidor utilizado, la versión del software de cada aplicación detectada.

El objetivo principal de una exploración de direcciones IP, es revisar todas las direcciones IP que conforman la red local, esto con el fin de encontrar una dirección IP que no esté siendo utilizada en ese instante. Lo cual significa que puede ser usada. Esta dirección IP puede servir para que el atacante sea capaz de hacerse pasar por un usuario legítimo. Por otro lado, una exploración de puertos revisa todos los diferentes puertos de la red, buscando algún puerto vulnerable. Este puede ser usado posteriormente por el atacante para alterar el funcionamiento normal de la red.

Las exploraciones de vulnerabilidad exploran las redes en busca de vulnerabilidades específicas. Un atacante que utiliza una exploración de vulnerabilidades recibirá una lista de las direcciones IP que son vulnerables a cierto tipo de ataque. Con este tipo de información el atacante puede identificar las máquinas dentro de la red con ataques específicos que pueden ser usados para penetrar a dicha red. Así los atacantes usan la exploración para detectar un objetivo específico antes de usar un ataque real.

Ataque de negación de servicio

Los ataques de negación de servicio (Nds) son ataques activos [19] los cuales intentan alentar o interrumpir por completo los servicios que presta un sistema o una red. Existen dos tipos principales de ataques de negación de servicio: *exploración de grietas o desperfectos e inundaciones*.

Exploración de grietas o desperfectos Una exploración de grietas utiliza un desperfecto en el software del sistema para causar una falla en los procesos o para agotar los recursos del sistema. Un ejemplo de este tipo de ataques es el “ping de la muerte”. Este ataque funciona mandando una inesperada gran cantidad de paquetes a cierto sistema con un sistema operativo Windows. El sistema no puede manejar la cantidad anormal de paquetes, y el resultado es un bloqueo del sistema. Con respecto a los ataques que intentan agotar los recursos del sistema, los recursos incluidos en el ataque incluyen: tiempo de procesador, memoria, espacio en disco, espacio en un buffer específico, o ancho de banda de una red. En muchos casos, el mantener actualizado el software puede prevenir este tipo de ataques.

Ataques de inundación Este tipo de ataques simplemente envían a un sistema o a un componente del sistema más información de la que puede manejar. En los casos en la que el atacante no puede mandar al sistema suficientes datos para inundar su capacidad de procesamiento, el atacante puede sin embargo ser capaz

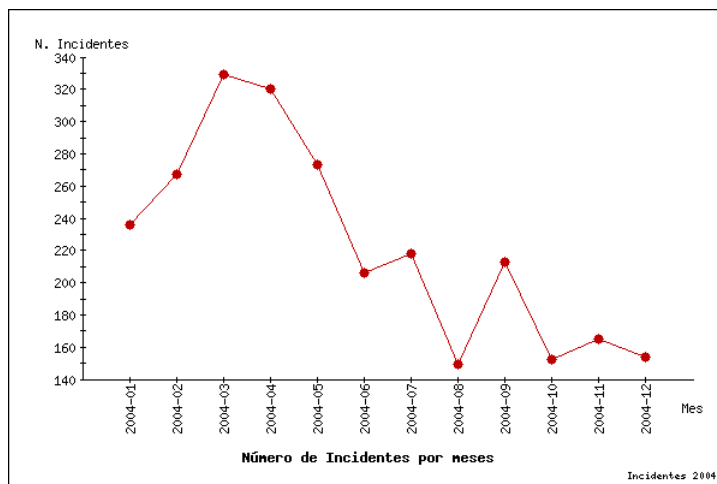


Figura 2.1: Número de incidentes por meses

de monopolizar la conexión al sistema, negando a alguien más el uso del recurso. Con este tipo de ataques, no existe ninguna grieta o desperfecto en el sistema que se deba reparar.

El termino “ataque de negación de servicio distribuido” (NdSD) es un subconjunto de NdS. Los ataques del tipo NdSD son simplemente ataques de inundación donde el atacante usa múltiples computadoras para ejecutar el ataque. Estas computadoras son controladas por la computadora del atacante para actuar como un inmenso sistema de ataque. Un atacante no puede agotar los recursos de un sistema o una red por sí solo. Sin embargo, si el atacante obtiene el control de 20,000 computadoras y las dirige todas para atacar la red, entonces el atacante obtiene una capacidad formidable para atacar satisfactoriamente hasta los sistemas y redes más rápidas.

En el año de 2004 Red-IRIS [34] reportó 2682 problemas de seguridad, entre los más comunes se encuentran las exploraciones de red, gusanos, troyanos DoS, entre otros. En la Figura 2.1 se muestra la gráfica de distribución de incidentes durante el año 2004 a lo largo de los diferentes meses.

En la Figura 2.2 se muestra la gráfica con una distribución de incidentes según su tipo.

Como se puede ver existe un dominio de los ataques de exploración. La característica principal detrás de una exploración es que detrás se encuentra un problema mayor y la causa de la misma.

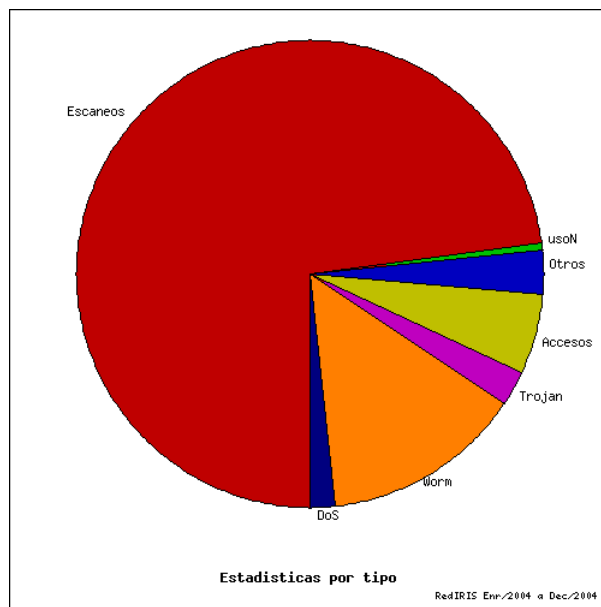


Figura 2.2: Incidentes por tipo

2.4. Sistemas de detección de intrusos

Los mecanismos de seguridad existentes protegen las redes de usos no autorizados a través de controles de acceso. Sin embargo, si estos controles son comprometidos o pueden ser burlados, un usuario puede ganar acceso o privilegios no autorizados y causar daño a la operación normal de la red.

A pesar de que la defensa principal de las redes son los controles de acceso, estos no son capaces de proteger la red completamente de intentos de intrusión, virus y robos computarizados. Por lo que no es posible depender de los mecanismos de control de acceso en todos los casos para estar protegidos contra una intrusión, o un ataque interno. Incluso las redes más seguras son vulnerables a abusos de usuarios internos, quienes hacen un mal uso de sus privilegios.

Un intruso por lo general presenta un patrón de comportamiento muy diferente del que corresponde a un usuario legítimo. Un intruso que se hace pasar por un usuario legítimo se puede detectar a través de la observación de su comportamiento inusual. Esta idea es la base para mejorar un sistema de seguridad monitoreando su actividad para detectar comportamientos atípicos.

A pesar de que muchos sistemas de supervisión recolectan datos sobre el comportamiento del tráfico de la red, muchos de ellos no tienen la capacidad de llevar un análisis

automatizado de los datos recogidos. Por otra parte, los sistemas que recolectan datos sobre el tráfico de la red, muchas veces recolectan grandes volúmenes de información que no necesariamente son relevantes para la seguridad. El gran volumen de los datos hace imposible para el administrador de la red detectar las actividades sospechosas, por lo tanto, es necesario tener la capacidad para analizar de manera automática el comportamiento del tráfico de la red.

La detección de intrusos es el proceso de monitorear los eventos que ocurren en un sistema o una red y analizarlos para detectar señales de *intrusiones*, las cuales son definidas como intentos de comprometer la confidencialidad, integridad y disponibilidad, o de violar los mecanismos de seguridad de la red. Las intrusiones son causadas por atacantes que quieren acceder a la red local desde Internet, por usuarios autorizados que intentan obtener privilegios adicionales para los cuales no están autorizados, y por usuarios autorizados quienes abusan de los privilegios dados a ellos. Los sistemas de detección de intrusos (SDI) son componentes de hardware o software que automatizan este proceso de monitoreo y análisis.

La detección de intrusos permite a las organizaciones proteger sus sistemas de las amenazas que aparecen al incrementar la conectividad entre las redes. Los sistemas detectores de intrusos han ganado la aceptación como un recurso necesario para la infraestructura de seguridad de cualquier organización.

2.4.1. Clasificación general

Lo visto hasta ahora permite realizar una clasificación de los sistemas detectores de intrusos según diversos criterios. Aunque hay más formas de clasificar estos sistemas, ésta es la manera más común. La Figura 2.3 muestra la clasificación de los principales tipos de sistemas detectores de intrusos.

2.4.2. Fuentes de información

La fuente de datos es una de las primeras cosas a tener en cuenta a la hora de diseñar un sistema de detección de intrusos. Estas fuentes se pueden clasificar de muchas maneras. En lo que respecta a la detección de intrusiones, se pueden dividir en cuatro categorías: *huésped, red, aplicación y objetivo*. Usaremos el término “monitorear” como el acto de recoger datos de una determinada fuente y enviarlos a un motor de análisis. *Monitores basados en máquinas (host based)*: Recogen los datos generados por

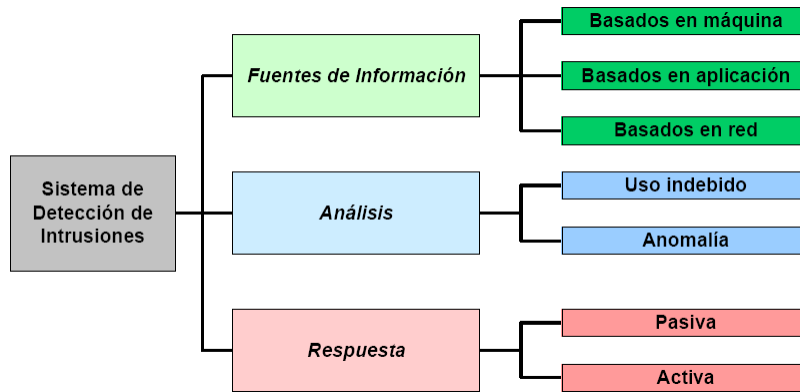


Figura 2.3: Tipos principales de sistemas detectores de intrusos

una computadora, normalmente a nivel del sistema operativo. Los registros de sucesos y las colas de auditoría pertenecen a este grupo [32, 37]. *Monitores basados en múltiples máquinas (multi-host based)*: Como su nombre lo indica, utiliza la información recogida en dos o más máquinas. Su enfoque es muy similar al basado en máquinas con la dificultad añadida de tener que coordinar los datos de varias fuentes. *Monitores basados en redes (network based)*: Capturan paquetes de red. Para ello, normalmente se utilizan dispositivos de red en modo promiscuo, convirtiendo al sistema en un rastreador o *sniffer*. *Monitores basados en aplicación (application based)*. Registran la actividad de una determinada aplicación. Por ejemplo, los registros de un servidor FTP. *Monitores basados en objetivos (target based)*: Estos monitores difieren ligeramente del resto porque generan sus propios registros. Utilizan funciones de cifrado para detectar posibles alteraciones de sus objetivos, y contrastan los resultados con las políticas de seguridad. Este método es especialmente útil cuando se usa contra elementos que, por sus características, no permiten ser monitorizados de otra forma. *Monitores híbridos (hybrid)*. Combinan dos o más fuentes de distinto tipo. Cada vez es más frecuente encontrarse con productos de detección de intrusiones basados en este punto de vista. Así, amplían sus posibilidades de detección.

Hay que añadir que existen sistemas de detección de intrusos que reciben el nombre de NNID (*Network Node Intrusion Detector*), es decir *Detector de Intrusos de Nodo de Red*. En realidad, son un caso especial de la detección basada en red. Este nombre se aplica cuando el monitor basado en red se sitúa en un *huésped*, monitorizando los paquetes destinados u originados por la máquina anfitriona. Una de las razones más importantes para hacer esto es para sortear el problema de las encriptaciones durante la

comunicación, ya que el tráfico es cifrado o descifrado por el propio huésped. Un detector basado en red convencional no podría analizar el tráfico en un punto intermedio entre dos nodos que cifrarán sus comunicaciones. Los NNIDS son también útiles en entornos de red con conmutadores en los que un huésped, aunque esté en modo promiscuo, sólo percibe el tráfico destinado a él.

2.4.3. Tipos de análisis

Después del proceso de recopilación de información, se lleva a cabo el proceso de análisis. La detección de intrusos también se puede clasificar según los objetivos del motor de análisis. Los dos tipos principales de análisis son:

Detección de usos indebidos

Un detector de usos indebidos es, a grandes rasgos, un comparador de patrones. Para que funcione correctamente necesita: una base de conocimiento con patrones fiables, una serie de eventos para poder ser analizados, y un motor de análisis eficaz. La detección de usos indebidos, por su propia naturaleza, tiene una limitación importante. Solo puede identificar problemas de seguridad cuando ya se han definido en su base de conocimiento. Esto significa que se han de conocer de antemano los métodos y ataques utilizados. La detección de usos indebidos se puede implementar de las siguientes formas:

Sistemas expertos. Los sistemas expertos utilizan reglas “if-then-else” para examinar los datos. Realizan análisis mediante funciones internas al sistema, de forma completamente transparente al usuario. Una de las ventajas más importantes de utilizar reglas “if-then” es que mantiene separados el control de razonamiento y la formulación de la solución del problema. En lo que respecta a los usos indebidos, permite deducir una intrusión a partir de la información disponible. La principal desventaja que se plantea es que los patrones no definen un orden secuencial de acciones. Detectar mediante este método ataques compuestos por una sucesión de eventos encierra grandes dificultades.

Sistema basado en modelos. Este sistema que consiste en una variación de la detección de usos indebidos, fue propuesto por T.D. Garvey y T. Lunt.[11] Utiliza una base de datos de escenarios de ataque, en la que cada escenario consiste en una secuencia de comportamientos que forman el ataque. En cada momento, el sistema analiza subconjuntos de comportamientos de su base de datos que coincidan con

los que está experimentando en ese instante. Utiliza los resultados para identificar y anticipar posibles ataques (*anticipador*). El anticipador determina los posibles comportamientos hostiles y las envía al *planificador*. Este planificador determina el grado de acierto entre el comportamiento recibido y el que figura en los registros de auditoría, y traduce los resultados en registros de auditoría del sistema.

La ventaja de este modelo radica en que está basado en una teoría matemática que utiliza el principio de incertidumbre. El diseño del *planificador* es independiente de la sintaxis del registro de auditoría. Además de esto, esta solución consume poco tiempo de proceso por cada registro de auditoría generado. Un inconveniente que presenta este modelo es que depende de la buena pericia del administrador de red en el momento de diseñar patrones creíbles y precisos. Por otra parte, el modelo no especifica claramente la forma en que se deben compilar los comportamientos en el planificador para que sea más eficiente, lo que afecta el rendimiento del detector.

Detección de anomalías

En la detección de anomalías se utilizan diversos algoritmos para crear perfiles de comportamiento normal, que sirvan de modelos a contrastar con la conducta actual de cada usuario [8, 18]. Las desviaciones que resultan de esta comparación son sometidas a técnicas que utiliza el sistema para decidir si ha habido o no indicios de intrusiones. Los *perfiles* están compuestos por conjuntos de *métricas*. Las métricas son medidas sobre aspectos concretos del comportamiento del usuario. El principal inconveniente de la detección de anomalías es que los perfiles de conducta pueden ser gradualmente *educados*. Un atacante que sepa que sus actividades están siendo monitorizadas, puede cambiar paulatinamente su forma de comportamiento a lo largo del tiempo, para que cuando cometa una intrusión no sea reconocida como tal. Esta técnica es conocida como “session creep” (deslizamiento, o movimiento sigiloso de sesión). Por otra parte, una de las ventajas consiste en su posibilidad de descubrir nuevos ataques, ya que se adapta y aprende de la conducta del sistema, al contrario de la detección de usos indebidos.

Modelo de Denning Dorothy Denning indica en su documento al menos cuatro modelos estadísticos que deben estar presentes en el sistema. Cada uno diseñado para adoptar un determinado tipo de medición [8]. Estos modelos son:

- Modelo Operacional: Se aplica sobre eventos para determinar por ejemplo, el número de intentos de accesos fallidos al sistema. Compara su medición con un

valor de umbral, que normalmente le indica si se ha cometido un intento de intrusión.

- Modelo de desviación media y estándar: Este modelo aplica el concepto de desviación media y estándar típico al momento de elaborar los perfiles de comportamiento. Supone que a partir de las dos primeras medidas, se puede establecer un intervalo de confianza. Este intervalo sirve para determinar la desviación estándar. Si las siguientes medidas caen fuera de este intervalo revelarían un comportamiento anormal.
- Modelo multivariable: Es una ampliación del modelo de desviación media y estándar. Utiliza correlaciones entre dos o más métricas para definir un comportamiento. Por ejemplo, en vez de determinar la conducta de un usuario exclusivamente por la duración de su sesión, también se tiene en cuenta la actividad de tráfico de red que genera durante la misma.
- Modelo del proceso Markov: Este es el modelo más complejo de los cuatro. Considera cada evento como una variable de estados, y utiliza una matriz de transiciones de estados para describir la frecuencia de transiciones de estados acumulados. Un evento determinado indica una anomalía si su probabilidad es demasiado baja. Este modelo ayudó a desarrollar análisis basados en *flujos de eventos* o la generación de patrones probables (*predictive pattern generation*).

Análisis cuantitativos Probablemente, las técnicas basadas en análisis cuantitativos son las más usadas para la detección de anomalías [18]. Mediante este método, las reglas de detección y los atributos de los objetos se expresan en forma numérica. Estos elementos se utilizan en análisis de diverso grado de complejidad. Los resultados se pueden usar para aportar patrones para elaborar perfiles relativos a la detección de anomalías. A continuación se describen algunos ejemplos de este tipo de análisis.

- Detección de umbral: Este sistema es conocido también como detección de umbral y disparador (*threshold and trigger*). En este caso, se cuenta el número de veces que ocurren los elementos que forman los perfiles de cada usuario, y se comparan estos datos con valores de umbral. Por ejemplo, si un usuario ha intentado entrar en el sistema más de cinco veces seguidas con una contraseña incorrecta, hay probabilidad de intrusiones. O si se realiza una serie de intentos de apertura

de puertos en un intervalo de tiempo muy corto, es posible que esté intentando explorar el sistema.

- Chequeos de integridad basados en objetivo: Consiste en utilizar alguna técnica que permita determinar si el objetivo monitorizado ha cambiado. Normalmente se utilizan funciones de resumen criptográfico (*hash*) sobre los objetivos, para generar sumas de control (*checksum*) almacenadas en una base de datos protegida. Si el objetivo en cuestión ha cambiado, su suma de control cambiará si este cambio es sustancial con respecto al anterior.

Medidas estadísticas Las medidas estadísticas constituyen uno de los primeros métodos utilizados para la detección de anomalías. Suelen servirse de *filtros de anomalías de protocolo* [9], que en realidad son filtros de anomalías estadísticas, adaptados para trabajar con protocolos de red.

El término *filtro de anomalías estadísticas*, se utiliza para definir sistemas que buscan eventos cuyo valor estadístico es inusualmente bajo, haciéndolos sospechosos. Un filtro de anomalías de protocolo, es un tipo de filtro de anomalías estadísticas al que se le ha añadido información específica sobre protocolos de red. Algunos de los sistemas que utilizan medidas estadísticas están basados en frecuencia, en los que la frecuencia con la que se da un determinado evento determina su probabilidad.

Sistemas basados en reglas Otra de las aproximaciones de la detección de anomalías es la basada en reglas (*rule-based*). Utiliza los mismos métodos que la detección de anomalías estadística. La diferencia radica en que la detección de intrusiones basada en reglas hace uso de conjuntos de reglas para representar y almacenar patrones de uso.

Redes neuronales Una de las formas de detección de anomalías más recientes es la basada en redes neuronales. El concepto consiste en aprovechar las características de aprendizaje de estas redes, para predecir las acciones de los usuarios, dado un número determinado n de acciones previas conocidas [11].

La red neuronal es entrenada mediante conjuntos de los comandos más significativos. Después de un período de entrenamiento, la red contrasta los comandos obtenidos con el perfil de usuario. Cualquier comando o acción predicha incorrectamente permiten determinar el grado de desviación entre el usuario y el perfil establecido. Las redes neuronales tienen la ventaja de que funcionan bien con

datos con ruido. No hacen suposiciones estadísticas sobre la naturaleza de los datos. Son capaces de detectar nuevas formas de ataque no conocidas, sin necesidad de reglas introducidas manualmente. Además, son fáciles de modificar para soportar nuevos conjuntos de usuarios [13].

Uno de los inconvenientes de estos sistemas está relacionado con la cantidad de datos a utilizar durante el entrenamiento. Las redes neuronales provocarían muchos falsos positivos si los datos son insuficientes, y son demasiados, el número de datos irrelevantes sería alto, aumentando los falsos negativos. Otro de los puntos en contra de este modelo es que no aporta ninguna explicación sobre las anomalías que identifica, dificultando la posibilidad de corregir las raíces del problema de seguridad en el sistema. Por otra parte, un intruso podría ser capaz de entrenar la red durante la fase de aprendizaje. Se han hecho estudios de soluciones híbridas que intentan solventar algunos de estos inconvenientes [5].

2.4.4. Tipos de respuesta a incidentes

El mecanismo de respuesta es otro de los factores que ayudan a definir el tipo de sistema de detección de intrusos. Los resultados obtenidos de la fase de análisis, se utilizan para tomar las decisiones que conducirán a una respuesta. Ésta es la tercera y última fase del modelo de un sistema de detección de intrusos. El conjunto de acciones y mecanismos que se pueden efectuar en esta etapa es amplio.

Las respuestas de un sistema de detección de intrusos pueden ser de dos tipos: *pasivas* o *activas* [19]. Las pasivas consisten en la emisión de informes, o el registro de las intrusiones ocurridas. Las activas son las que implican alguna acción en particular, como el bloqueo de conexión, el cierre inmediato de una cuenta, o la prohibición de ejecución de determinados comandos.

Se pueden soportar ambos tipos de respuesta en un sistema de detección. Una no excluye a la otra. Es posible que en determinadas anomalías sólo sea necesario registrar la actividad ocurrida para su posterior examen, mientras que en intrusiones a sistemas críticos haga falta una actuación más activa y urgente.

Respuestas activas

Las respuestas activas, como ya se mencionó, afectan al progreso del ataque, pueden ser llevadas a cabo de forma automática por el sistema, o mediante intervención humana.

Estas acciones pueden ser de diversas naturaleza; no obstante, la mayoría se pueden clasificar en estas tres categorías principales: ejecutar acciones contra el intruso, corregir el entorno, y recopilar información.

Ejecutar acciones contra el intruso La forma más directa consiste en identificar el origen de ataque, e impedir el acceso al sistema. Por ejemplo, desactivando una conexión de red, o bloqueando la máquina comprometida. Sin embargo, tomar decisiones tan agresivas no siempre es una buena solución. Hay situaciones en las que esto podría causar serios problemas:

- Los ataques recibidos a menudo son realizados, no desde la propia máquina del intruso, sino desde una víctima controlada remotamente. Si se bloquea o incluso devuelve el ataque en esta situación, se estaría perjudicando al usuario.
- En muchas ocasiones, los atacantes utilizan técnicas de ocultación de dirección IP (spoofing). En este tipo de ataques, las direcciones IP de origen no tienen nada que ver con el atacante. Incluso pueden no existir, hecho beneficioso en casos de ataques de negación de servicio, en los que el servidor pierde demasiado tiempo esperando la respuesta de direcciones IP que no responden nunca.

Corregir el entorno Esta opción como su nombre indica, consiste en efectuar acciones pertinentes para restaurar el sistema y corregir los posibles problemas de seguridad existentes. En la mayoría de las ocasiones, esta respuesta activa suele ser la acción más acertada. Aquellos sistemas que cuentan con métodos de *auto-curación* (self-healing), son capaces de identificar el problema y proporcionar los métodos adecuados para corregirlo. En muchas ocasiones, este tipo de respuestas puede provocar cambios en el motor de análisis, generalmente aumentando su sensibilidad e incrementando el nivel de sospecha ante posibles intrusiones.

2.4.4.1. Respuestas pasivas

Son aquellas respuestas que consisten en el envío de información al administrador de la red, dejando recaer en él la toma de decisiones. Las alarmas por pantalla son una de las alarmas más comunes entre los sistemas de detección de intrusos. Un mensaje en una ventana indica al usuario que se ha cometido una posible intrusión, acompañando el mensaje con información adicional, como la dirección del posible atacante, el protocolo usado, etc.

Otra de las posibles formas de recibir respuestas pasivas es a través del correo electrónico el cual puede contener más información, proporcionando un mensaje más extenso, y menos ambiguo, sobre la incidencia.

Por otra parte algunos sistemas de detección de intrusos están integrados con mecanismos de gestión de redes. En estos casos se suele hacer uso de mensajes SNMP (*Simple Network Management Protocol*). La integración con este sistema de comunicación permite utilizar canales ya existentes para el envío de incidencias. No obstante, un uso excesivo de estos canales, podría perjudicar a otros sistemas que también hicieran uso de ellos.

2.4.4.2. Falsas alarmas

Uno de los problemas asociados a los mecanismos de respuesta de los sistemas detectores de intrusos es el de las falsas alarmas. Son de dos tipos: *falsos positivos*, que indican posibles intrusiones cuando en realidad no los hay; y *falsos negativos*, que no notifican intrusiones cuando realmente han tenido lugar.

Los falsos negativos representan un problema, en el sentido de que al sistema se le escapan muchas actividades sospechosas. La solución para este problema es mejorar la sensibilidad del detector, afinando la configuración o desarrollando mejores técnicas de detección.

Los falsos positivos pueden ser también peligrosos si se producen con demasiada frecuencia. Pueden hacer que el administrador de la red acabe ignorándolos, o no distinguiendo entre ellos las verdaderas alarmas. En el peor de los casos, pueden llegar a colapsar el sistema. Una de las posibles soluciones a esta situación consiste en disminuir la sensibilidad del sistema detector de anomalías, adaptando los resultados al comportamiento normal de la red.

2.5. Medición del tráfico

Cuando se está analizando el desempeño y comportamiento de una red, la medición de la misma es necesaria. Para realizar una buena medición del tráfico es necesario decidir que parámetros son de nuestro interés. El parámetro más significativo es el número de paquetes que pasan a través de la red en un tiempo determinado [4, 10]. El número de paquetes tiene varias ventajas ya que es fácil de estimar y es independiente de la tecnología y capas de red. *Medición activa*: La medición activa esta basada en la inyección

de tráfico dentro de la red, con el propósito de medir uno o varios parámetros basados en el comportamiento que presente el tráfico agregado. Sin embargo, la medición activa puede generar datos falsos de ciertos parámetros o incluso afectar el comportamiento y desempeño de la red al sobrecargarla. *Medición pasiva:* Las mediciones pasivas están basadas en la observación del tráfico ya existente en la red. Esto es debido a que no se añade tráfico adicional a la red. La ventaja principal es que los parámetros bajo estudio no serán modificados por la medición. La medición pasiva es usada en gran medida cuando la red que se va a medir es grande, ya que este tipo de medición no genera tráfico extra en la red. Por otro lado, dependiendo del nivel de detalle deseado, la medición activa puede generar una cantidad enorme de datos, los cuales pueden ser un problema al momento de almacenarlos. Además, un problema importante es la privacidad de los datos. Ya que como el tráfico de la red es copiado, es posible leer los datos contenidos. Esto requiere que los datos capturados sean tratados con cuidado. Solo se debe almacenar los datos útiles para la medición, sin desechar información relevante. Este es un problema que no tiene la medición activa.

2.6. Modelado de tráfico

Los modelos de tráfico de red son ampliamente usados en el análisis de comportamiento de la red. Los usos más frecuentes de los modelos son la simulación de tráfico y el análisis de desempeño de la red. Un modelo de tráfico intenta predecir o reproducir artificialmente el comportamiento de los paquetes que circulan en una red [10].

Existen varios factores que dificultan la realización de un modelo que reproduzca completamente el comportamiento natural de una red. Entre estos factores se encuentran los siguientes: La rápida evolución de la redes, los diferentes protocolos utilizados en la red, la tecnología usada por la red (ATM, Token Ring, Ethernet, etc.), los diferentes dispositivos de interconexión usados, el comportamiento aleatorio presentado por los usuarios.

Un modelo estadístico es aquel que se basa en un análisis estadístico del comportamiento previo de un evento para poder determinar de manera aproximada el valor que tendrá en futuras repeticiones. Los modelos estadísticos usan una gran cantidad de datos de comportamiento obtenidos previamente. Los modelos estadísticos no son tan precisos como los modelos matemáticos sin embargo, un modelo estadístico bien planificado, nos puede ayudar a reproducir el comportamiento de una red local de una

manera rápida y lo suficientemente cercana al comportamiento real.

Los modelos estadísticos se pueden clasificar en dos tipos: *estáticos y dinámicos*. Un modelo estadístico estático es obtenido a partir de observaciones previas. El modelo estadístico estático no tendrá modificaciones futuras en tiempo de ejecución. Por otro lado, un modelo estadístico dinámico es obtenido de la misma manera que un modelo estático, sin embargo este tiene la capacidad de adaptarse de manera automática a los cambios normales que sufre una red local. Cada vez que la red sufre un cambio normal, un modelo actualizado es obtenido de manera automática. Esto implica que un modelo estadístico dinámico debe ser capaz de distinguir entre un comportamiento normal y un ataque o anomalía en la red.

En resumen hemos mostrado la arquitectura de las redes de área local, así como la manera en que estas se comunican entre sí por medio de los diferentes dispositivos de interconexión existentes. Al estar comunicadas, las redes se encuentran expuestas a ataques en busca de obtener o alterar la información que circula a través de ellas. Debido a esto, es necesario llevar a cabo la medición del tráfico que circula por las mismas. Los sistemas detectores de intrusos intentan encontrar comportamientos anómalos en el tráfico de la red sin embargo, como hemos visto la mayoría de los comportamientos anómalos son originados por una exploración de red no detectada a tiempo. En el siguiente capítulo mostraremos una manera sencilla y de bajo costo para la detección de comportamientos anómalos mediante el uso de modelos estadísticos dinámicos y la búsqueda de patrones de exploraciones en el tráfico de red.

Capítulo 3

Supervisión automática de redes locales

El incremento en el tamaño de las redes locales actuales ha dificultado en gran manera la detección de comportamientos anómalos por parte de los administradores de red ya que es necesario navegar a través de múltiples datos sobre el tráfico de red en busca de información. Una herramienta útil para el análisis de los datos son los *sniffers* [36] los cuales son programas capaces de monitorizar el tráfico circulante de una red local y proporcionar datos sobre cada paquete de red existente.

Actualmente el CINVESTAV cuenta con un sistema monitor de tráfico de red [10] el cual proporciona datos sobre los paquetes que circulan a través de una red local. Sin embargo, el análisis de estos datos debe efectuarse de manera manual por el administrador de red, lo que es prácticamente una tarea imposible de realizar continuamente, por este motivo es necesario un método para automatizar la tarea de detección de comportamientos anómalos en el tráfico de red.

Mediante algunas extensiones al sistema desarrollado, se construye una herramienta eficaz y de bajo costo para la detección automática de comportamientos anómalos en una red de área local. El sistema propuesto se divide en tres partes. La primera consiste en la obtención y filtrado de los datos de tráfico de red almacenados en la base de datos del sistema monitor, para ser utilizados por las extensiones desarrolladas. La segunda parte consiste en el uso de dichas extensiones para efectuar un análisis automático del tráfico de red para obtener datos estadísticos sobre la misma. La tercera parte utiliza los datos del análisis para realizar una detección automática de comportamientos anómalos en el tráfico de una red de área local. Cada una de las partes se explica en las secciones

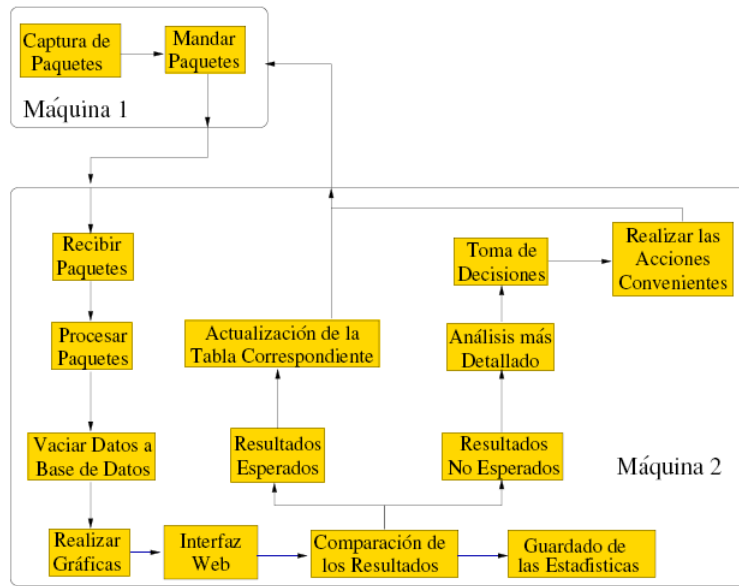


Figura 3.1: Diagrama del sistema monitor de tráfico de redes locales.

3.1, 3.2, 3.3 respectivamente.

3.1. Sistema monitor de tráfico de redes locales

En esta sección se explica el trabajo desarrollado por Jorge Enrique Morfín Galván en su tesis de maestría titulada “Análisis del Tráfico de una Red Local” [10]. La tarea del sistema monitor de red consiste en capturar todos los paquetes entrantes y salientes que circulan a través de una red de área local, almacenar la información del encabezado de cada uno de ellos en una base de datos, usar dichos datos para realizar gráficas que son mostradas por medio de una interfaz Web y realizar un análisis estadístico.

El sistema monitor de tráfico de red utilizado se encuentra constituido por tres módulos principales como se muestra en la Figura 3.1¹, el módulo de captura de tráfico, una interfaz Web y un modelo estadístico estático. Estos módulos se encuentran distribuidos en dos computadoras, en la primera se encuentra el módulo de captura de tráfico, mientras que en la segunda se encuentran los otros dos módulos.

Existen varios métodos para efectuar la captura de paquetes de red los cuales se pueden clasificar en pasivos y activos [36] los cuales a su vez pueden estar basados en

¹Imagen tomada de [10].

hardware y software [4, 45]. El módulo de captura de tráfico utiliza un método basado en software efectuando la captura de paquetes de manera pasiva, esto le proporciona la ventaja de no requerir de hardware especializado para realizar la captura de paquetes además de no generar ningún tipo de tráfico adicional hacia la red. El módulo de captura hace una copia de los datos del encabezado de cada paquete que circula por la red enviándola a la segunda computadora para su almacenamiento. La información enviada incluye la dirección IP fuente y destino, puerto fuente y destino y protocolo utilizado por cada paquete de red. Sin embargo, aunque esta información revela datos valiosos sobre el comportamiento general de la red, esta no incluye información alguna sobre la topología de la misma.

Una vez que los datos sobre los paquetes de red son enviados por el módulo de captura estos son recibidos por la segunda computadora para su almacenaje en una base de datos. Utilizando los datos disponibles en la base de datos son realizadas gráficas sobre el comportamiento general de la red para su visualización en una interfaz Web. Las gráficas presentadas en la interfaz incluyen el tráfico de las últimas 24 horas de comportamiento, porcentajes sobre protocolos utilizados, número de paquetes por dirección IP, comparación del modelo estadístico estático [10] contra el tráfico general circulante. Mediante estas gráficas es posible llevar a cabo un análisis sobre el desempeño general de la red en busca de fallas, ataques y abusos de gran magnitud. Sin embargo, debido a que este análisis debe efectuarse de manera visual es necesario efectuar numerosas revisiones al día, aunado a esto el gran número de datos que deben ser analizados hacen de ésta una tarea ardua. Más aún, ya que las gráficas sólo muestran el comportamiento general de la red, no es posible detectar comportamientos anómalos de pequeñas magnitudes.

El modelo estadístico estático proporcionado por el sistema monitor de tráfico fue obtenido con el propósito de predecir el comportamiento general de la red en un lapso de tiempo determinado. Este modelo consiste en el almacenamiento del promediado del número de paquetes que circulan a través de la red en lapsos de 10 minutos. Dicho modelo fue obtenido mediante la recolección, promediado y suavizado de una gran cantidad de datos correspondientes al número de paquetes circulantes por la red de área local correspondiente a varios días de uso continuo. El modelo estadístico obtenido ofrece una guía para determinar el nivel del desempeño general de la red de área local sin embargo, debido a que solo se cuenta con un solo modelo estadístico para predecir el comportamiento de la red de todos los días de la semana esto se traduce en un bajo nivel de exactitud de las estadísticas debido a que la cantidad de tráfico circulante varía dependiendo el día de la semana siendo los casos más notables los fines de semana

o los días no laborables. Aunado a esto, ya que el modelo es estático, este debe ser actualizado u obtenido a entero del administrador del sistema cuando sea efectuado un cambio en la arquitectura o configuración de la red de área local.

En resumen el sistema monitor de tráfico de redes locales proporciona información importante sobre el comportamiento de los paquetes circulantes, contando con las siguientes capacidades: proporcionar información del origen y destino de cada paquete de red circulante, mostrar gráficas sobre el comportamiento general de la red a través de una interfaz Web, comparación del comportamiento general de la red con el modelo estadístico estático, y cálculo del error entre el tráfico esperado por el modelo estático y el tráfico real obtenido de la red. Sin embargo, el sistema tiene ciertas limitaciones como la carencia de un modelo estadístico para cada día de la semana así como la necesidad de actualizar de manera manual dichos modelos cuando se presente un cambio en la arquitectura de la red. Otra limitación que presenta el sistema es la carencia de información de la topología de la red para, de esta manera aumentar la exactitud y profundidad del análisis de paquetes de red.

Por lo anterior, se proponen las siguientes extensiones para el sistema monitor de tráfico de redes locales.

- La actualización del modelo estadístico estático a varios modelos estadísticos dinámicos correspondientes a los siete días de la semana.
- La integración de la información de la topología de la red.
- Obtención de modelos estadísticos dinámicos por cada dispositivo de interconexión.
- Integración de detectores que sean capaces de utilizar los modelos dinámicos para la detección de algunos comportamientos anómalos.

En las secciones siguientes se describirán con mayor detalle cada una de estas extensiones.

3.2. Análisis automático del tráfico de red

En esta sección se explican las extensiones necesarias para el desarrollo de un modelo estadístico dinámico el cual tiene como objetivo servir como una base de comparación para la detección de anomalías en el comportamiento de la red. Además, se muestra la

adición de la información de la topología de red, la cual tiene como fin proporcionar un modelo detallado del comportamiento de la red actual.

3.2.1. Modelo estadístico dinámico

Existen varios métodos que pretenden reproducir de manera artificial el comportamiento de una red de área local con el propósito de contar con una base de comparación para la detección de comportamientos anómalos. Entre los últimos estudios se encuentran los modelos de auto-similitud [44, 28], lógica difusa [41] y modelos estadísticos [10].

La auto-similitud es una propiedad en la cual un todo se parece a cada una de sus partes y viceversa. Este principio es presentado por el tráfico de red ya que si se grafica un día entero de tráfico de red y a la vez se grafica solo unos minutos de éste, estas dos gráficas serían muy parecidas. Siguiendo este principio es posible entonces tratar de predecir el comportamiento de un día entero de tráfico de red utilizando pequeñas partes de éste como base de cálculo. Sin embargo, el costo computacional de tal procedimiento es muy alto y poco efectivo si lo que se desea es una respuesta en tiempo real.

Los modelos matemáticos basados en la lógica difusa son muy exactos para predecir el comportamiento que presentará el tráfico de red. Este modelo se basa en el modelo auto-regresivo utilizando intervalos de tráfico de red como muestra para predecir el tráfico que circulará durante los siguientes intervalos. Los resultados obtenidos con este modelo son muy exactos comparándolos contra el tráfico real presentado sin embargo, para llevar a cabo el cálculo de éste, es necesaria una enorme cantidad de datos así como una gran cantidad de tiempo, por lo que no son recomendables para una comparación de tráfico en tiempo real.

El modelo estadístico estático elaborado en [10] propone un método de bajo costo computacional, el cual es útil para comparaciones realizadas en un tiempo corto. El modelo estadístico estático, a pesar de carecer de un nivel alto de exactitud, resulta útil para la comparación de tráfico en un tiempo de respuesta corto sin embargo, al ser estático presenta la desventaja de tener que ser actualizado o cambiado al momento de que la arquitectura o configuración de la red cambie. Este problema se puede reducir al cambiarlo por un modelo estadístico dinámico.

Un modelo estadístico dinámico es similar a un modelo estadístico estático, con la ventaja de actualizarse a sí mismo de manera automática, de acuerdo a los cambios presentados en el comportamiento de la red. Más aún, con el propósito de mejorar el

nivel de exactitud del modelo estadístico, se añadió la capacidad de almacenar diferentes modelos estadísticos dinámicos de la red, los cuales, además de estar clasificados los modelos por día de la semana, también son clasificados por dispositivo de interconexión.

A continuación se explica la definición de los modelos estadísticos dinámicos desarrollados, así como los pasos necesarios para su actualización automática.

Un modelo estadístico dinámico varía de acuerdo al día, hora y dispositivo de interconexión como sigue:

$$f(d, h, c) = n \quad (3.1)$$

donde:

d es el día de la semana.

h es la hora y minuto del día.

c es el dispositivo de interconexión.

n es el número de paquetes esperados en una dirección de tráfico.

La Figura 3.2 ilustra un modelo estadístico del comportamiento del tráfico de entrada de una red de área local, en esta gráfica es ilustrado el número de paquetes circulantes esperados por el modelo estadístico en un día completo de uso de la red. Como se puede observar el ancho de banda utilizado por las noches es poco, pero el uso de éste aumenta durante el transcurso del día. Un modelo diferente se observa para el tráfico de salida.

Un modelo estadístico dinámico tiene la capacidad de actualizarse de manera automática, lo cual es posible mediante la distinción de cambios en la red es decir, el modelo debe ser capaz de distinguir entre un cambio normal en el comportamiento de la red y un cambio originado por una falla, ataque o abuso. Para lograr este objetivo es necesaria la definición de una desviación en el comportamiento de la red.

Una desviación en el comportamiento de la red puede ser interpretada de acuerdo a su magnitud y duración. Considerando la magnitud se determinará si el modelo debe o no ser actualizado, mientras que la duración determinará si existe o no un comportamiento anómalo en el tráfico de la red.

Para medir la magnitud de la diferencia entre el comportamiento esperado por el modelo y el real en un lapso de tiempo Δt , se calcula el error de la siguiente manera:

$$e(d, h, c) = \frac{|f_m(d, h, c) - f_r(d, h, c)|}{f_m(d, h, c)} \quad (3.2)$$

donde:

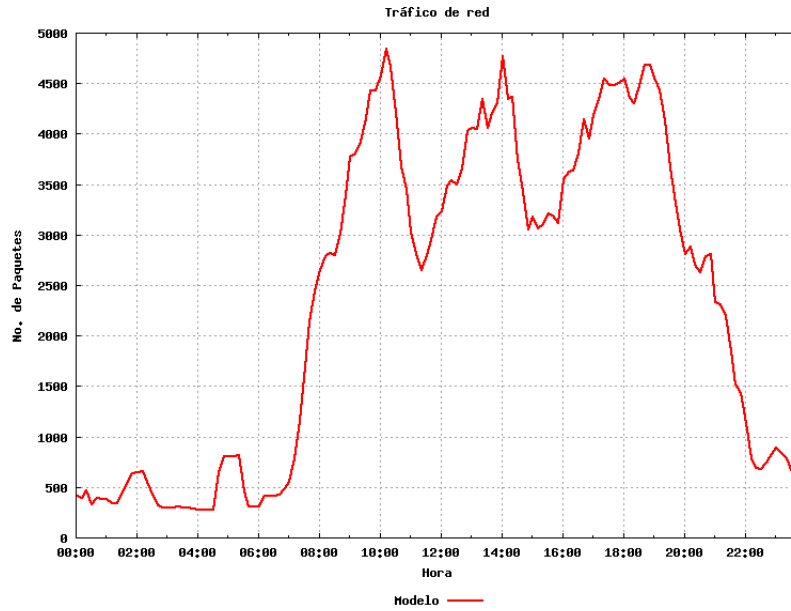


Figura 3.2: Modelo estadístico del comportamiento de una red de área local

$f_m(d, h, c)$ es el número de paquetes circulantes esperados por el modelo.

$f_r(d, h, c)$ es el número de paquetes circulantes reales.

Nótese que la definición de $e(d, h, c)$ considera el error relativo con respecto al valor esperado. El error esperado determina si un cambio en el comportamiento de la red se debe a un cambio normal ocasionado por la aleatoriedad normal presentada en una red de este tipo o si éste se puede deber a un comportamiento anómalo. Sea δ una tolerancia máxima de la desviación del comportamiento observado comparado con el esperado.

Cuando $e(d, h, c) < \delta$ entonces se entiende que la diferencia entre el número de paquetes circulantes reales y el número de paquetes esperados por el modelo fue mínima y no representa una amenaza para el desempeño de la red, por lo que el modelo puede ser actualizado utilizando los nuevos datos. Por el contrario, si $e(d, h, c) \geq \delta$ entonces la diferencia entre el comportamiento real y el esperado fue bastante grande por lo que esta se puede deber a un comportamiento anómalo en la red, en este caso el modelo no debe ser actualizado utilizando estos datos, ya que esto representaría un error en las estadísticas almacenadas. El valor δ utilizado fue de 0.5 ya que este ofrecía el mejor balance para la toma de decisiones al no generar un porcentaje alto de falsos positivos dejando pasar solo los datos útiles para el modelo.

La actualización automática del modelo estadístico se realiza mediante la aplicación

de la siguiente fórmula:

$$f_{new}(d, h, c) = \begin{cases} f_m(d, h, c) + ((f_r(d, h, c) * g) * m) & e(d, h, c) < \delta \\ f_m(d, h, c) & e(d, h, c) \geq \delta \end{cases} \quad (3.3)$$

$$m = \text{signo}[f_r(d, h, c) - f_m(d, h, c)]$$

donde:

g es un factor de modificación.

El valor g es de gran importancia ya que determina la intensidad con la que los nuevos datos afectarán a las estadísticas almacenadas en el modelo. Un valor g muy grande significaría un incremento o decremento muy brusco de los valores del modelo estadístico por lo que esto se vería reflejado en un nivel de falsas alarmas muy alto al momento de efectuar la comparación. Por el contrario un valor g demasiado bajo tiene como consecuencia una actualización muy lenta del modelo estadístico. Experimentalmente, se encontró un balance adecuado con $g = 0.05$.

En la Figura 3.3 se muestra el modelo estadístico dinámico (línea continua) comparado con el comportamiento real de la red (línea con puntos).

Haciendo uso de las formulas anteriores es posible determinar los segmentos $\Delta t_1 \dots \Delta t_n$ del modelo estadístico que son susceptibles de ser actualizados mediante el cálculo de la desviación $e(d, h, c)$ presentado por cada segmento. Este procedimiento asegura la correcta actualización continua del modelo estadístico, sin embargo, este dato por sí solo es insuficiente para la detección de comportamientos anómalos, debido a que es imposible determinar si el error obtenido durante un solo intervalo Δt fue causado por el comportamiento aleatorio de la red o se deba al principio de un comportamiento anómalo.

Para solucionar este problema se utiliza la duración del error como base para determinar su naturaleza, debido a que la duración de un error originado por un aumento o disminución en el tráfico de la red tiende a ser pequeña mientras que una falla tiene una duración mayor.

Es necesario entonces medir la duración que presenta un comportamiento que se encuentra fuera de los límites establecidos por $e(d, h, c)$. Para esto se hace uso del valor $\Delta e(d, h, c)$, el cual tiene como objetivo servir de límite para la clasificación entre un cambio normal en la red y un posible comportamiento anómalo. El valor de $\Delta e(d, h, c)$

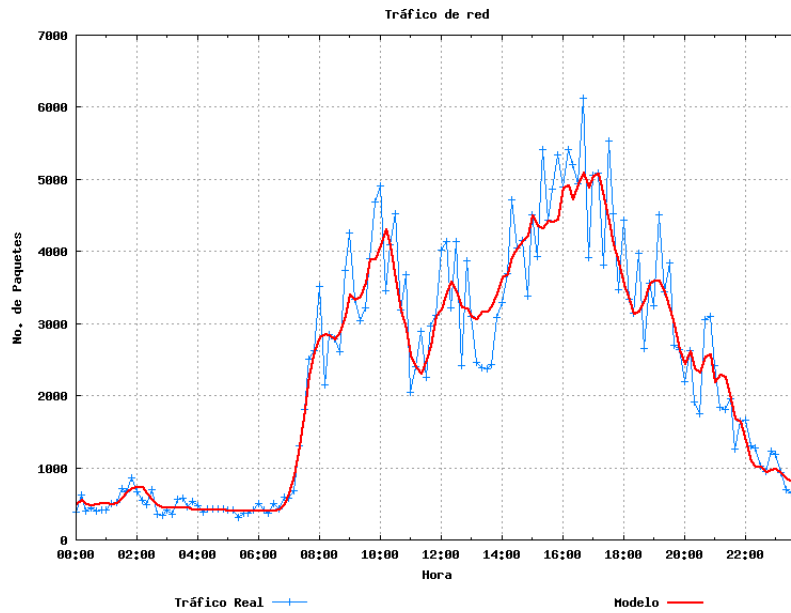


Figura 3.3: Modelo estadístico dinámico comparado con el comportamiento real de la red

determina el tipo y duración de las anomalías a detectar.

Cada intervalo $\Delta t_1 \dots \Delta t_n$ que rebasa la frontera establecida por $e(d, h, c)$ incrementará el valor $\Delta e(d, h, c)$ en una unidad, de esta manera es posible detectar intervalos $\Delta t_1 \dots \Delta t_n$ continuos que presenten un comportamiento diferente al establecido por el modelo. Cuando un intervalo Δt se encuentre dentro de los límites de $e(d, h, c)$ regresará el valor de $\Delta e(d, h, c)$ a cero. Mediante el uso del límite $\Delta e(d, h, c)$ es posible efectuar la detección de comportamientos anómalos estableciendo la duración de estos.

Existen diversos tipos de comportamientos anómalos que pueden ser detectados mediante el uso de métodos estadísticos, los cuales varían de acuerdo a su duración [29] como sigue: *Red*: Una falla en la red o una desconfiguración temporal dando como resultado una problema o una interrupción en el servicio. *Ataque*: Un aumento inusual del uso de ancho de banda, comúnmente originado por un ataque de Negación de Servicio. *Picos*: Un aumento del uso de ancho de banda de poca duración, por ejemplo, un usuario usando un servidor FTP.

Con el fin de implementar la detección automática de cada uno de ellos, se definió el nivel de tolerancia $\Delta e(d, h, c)$ para cada caso, esto es mostrado en el Cuadro 3.1. Los valores fueron obtenidos de manera experimental mediante la simulación de tráfico de red ocasionando deliberadamente cada uno de estos escenarios. Durante las pruebas

Falla	Duración
Picos	$5\Delta t \geq \Delta e(d, h, c) > \Delta t$
Red	$\Delta e(d, h, c) > 6\Delta t$
Ataque	$\Delta e(d, h, c) > 9\Delta t$

Cuadro 3.1: Nivel de tolerancia para comportamientos anómalos

realizadas estos niveles de tolerancia demostraron un nivel de detección aceptable con un rango de falsas alarmas muy bajo, alrededor del 9%.

En la Figura 3.4 se muestra la gráfica de error de un comportamiento de tráfico simulado en la que se muestra el número de paquetes de diferencia entre el comportamiento presentado y el modelo estadístico. Dentro del rectángulo gris se presenta un ataque simulado intencionalmente que da inicio desde las 14 horas, finalizando a las 18 horas del día. En el Cuadro 3.2 se muestra la evolución de $\Delta e(d, h, c)$ presentada en este lapso de tiempo.

Como se puede observar el valor $e(d, h, c)$ determina la magnitud del error entre el comportamiento real y el comportamiento esperado por el modelo consiguiendo de esta manera determinar la existencia de un comportamiento anómalo. Por otro lado el valor $\Delta e(d, h, c)$ determina la duración de los comportamientos anómalos detectados el cual puede ir desde un simple pico en el tráfico hasta un ataque en la red.

Con el uso de las fórmulas anteriormente mostradas se habilita la generación de un conjunto de modelos correspondientes a cada uno de los dispositivos de interconexión para cada día de la semana, así como el uso de la detección de cambios bruscos en el comportamiento del tráfico de red para realizar la actualización automática de los modelos estadísticos. A su vez se muestra una manera de detección de comportamientos anómalos con el uso de un nivel de tolerancia.

Existen comportamientos anómalos que no son susceptibles de ser detectados mediante métodos estadísticos, estos son explicados a detalle en la sección 3.3.

3.2.2. Topología de una red de área local

Para una mayor efectividad de los modelos estadísticos es necesario contar con la información correspondiente a la topología de la red supervisada, ya que es necesario conocer la carga de tráfico que circula por cada dispositivo de interconexión. La topología de una red es de gran utilidad para lograr este objetivo ya que ésta revela datos valiosos

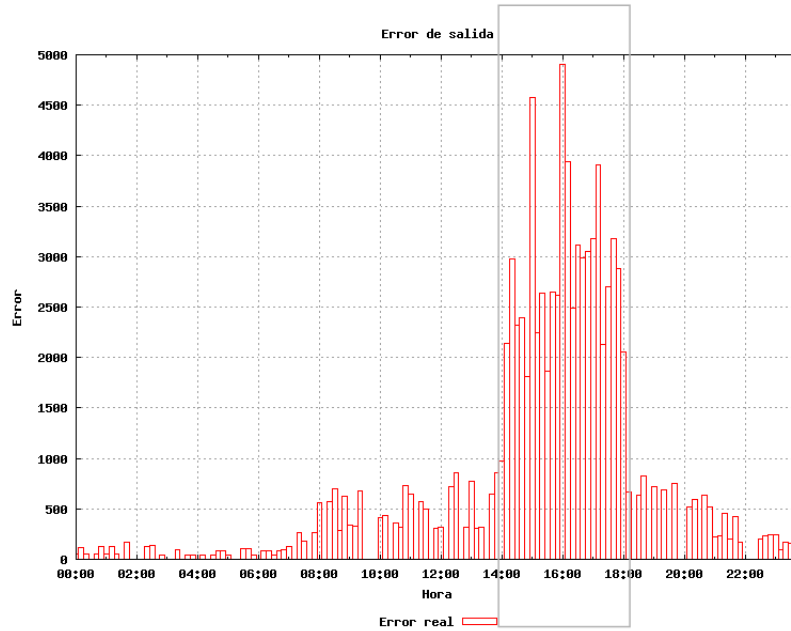


Figura 3.4: Gráfica de error del modelo estadístico dinámico

Hora	$\Delta e(d, h, c) - \Delta t$	$e(d, h, c) - \delta$	Comportamiento	Alarma
13:40	$\Delta e(d, h, c) = 0\Delta t$	$e(d, h, c) < \delta$	Normal	Nula
13:50	$\Delta e(d, h, c) = 0\Delta t$	$e(d, h, c) < \delta$	Normal	Nula
14:00	$\Delta e(d, h, c) = \Delta t$	$e(d, h, c) > \delta$	Normal	Nula
14:10	$\Delta e(d, h, c) = 2\Delta t$	$e(d, h, c) > \delta$	Pico	Baja
14:20	$\Delta e(d, h, c) = 3\Delta t$	$e(d, h, c) > \delta$	Pico	Baja
⋮				
14:50	$\Delta e(d, h, c) = 6\Delta t$	$e(d, h, c) > \delta$	Red	Media
15:00	$\Delta e(d, h, c) = 7\Delta t$	$e(d, h, c) > \delta$	Red	Media
15:10	$\Delta e(d, h, c) = 8\Delta t$	$e(d, h, c) > \delta$	Red	Media
15:20	$\Delta e(d, h, c) = 9\Delta t$	$e(d, h, c) > \delta$	Red	Alta
15:30	$\Delta e(d, h, c) = 10\Delta t$	$e(d, h, c) > \delta$	Ataque	Alta
⋮				
18:00	$\Delta e(d, h, c) = 25\Delta t$	$e(d, h, c) > \delta$	Ataque	Alta
18:10	$\Delta e(d, h, c) = 0\Delta t$	$e(d, h, c) < \delta$	Normal	Nula

Cuadro 3.2: Evolución del valor $\Delta e(d, h, c)$ del comportamiento mostrado dentro del rectángulo en la Figura 3.4.

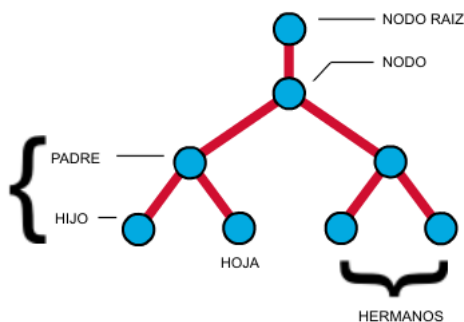


Figura 3.5: Partes de un árbol

acerca de la manera de cómo se encuentran conectados cada uno de los dispositivos que conforman una red como son: dispositivos de interconexión y computadoras.

Existen diferentes estructuras topológicas mediante las cuales se puede organizar una red, sin embargo, mediante árboles se cubre un alto porcentaje de las redes locales [35]. En esta sección se explican las principales características de una topología de árbol, la manera en la que los paquetes de red circulan a través de estas así como la forma en la que se utilizan los datos del tráfico de red y la topología para la obtención y uso en los modelos estadísticos dinámicos.

Una topología de árbol se asemeja a una estructura de datos del mismo nombre, en la cual a diferencia de un árbol natural la raíz del árbol se encuentra en la parte superior del mismo mientras que las hojas se encuentran en la parte inferior como se ilustra en la Figura 3.5 en esta, se muestran las partes más importantes de un árbol, así como también dos relaciones claves que existen entre los nodos de un árbol, la primera es la relación padre-hijo y la segunda es la relación de hermanos, esta última la presentan los nodos que son hijos del mismo nodo padre.

Una topología de árbol esta constituida por dos tipos de nodos: nodos padre y nodos hijo. Los nodos padre son los diferentes dispositivos de interconexión de red como son: concentradores, conmutadores y puertas de enlace, siendo los nodos hijo las computadoras de la red. Solo los dispositivos de interconexión pueden ser nodos padre, sin embargo, existe un único nodo que no tiene padre y es llamado el nodo raíz el cual, se localiza en la parte más alta del árbol [14], la raíz del árbol se puede considerar como la puerta de enlace de la red local.

Debido a la estructura jerárquica de una topología de árbol, el tráfico que circula por esta presenta un patrón bien definido: Todo el tráfico circulante entra o sale en su

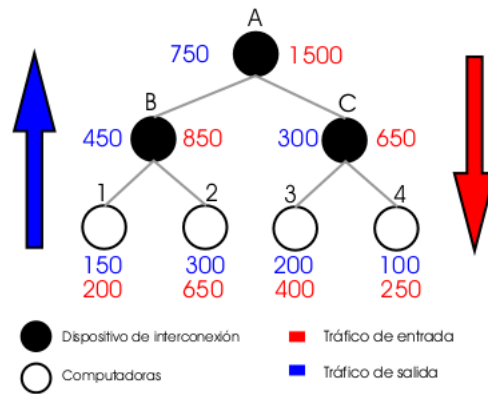


Figura 3.6: Tráfico de red en una topología de árbol

totalidad a través del nodo raíz de la red siendo distribuido hacia las diferentes computadoras por medio de los dispositivos de interconexión. Este patrón es ejemplificado en la Figura 3.6. Como se puede ver solo las computadoras son las generadoras del tráfico de red; el tráfico de salida es generado desde las computadoras y es distribuido por los dispositivos de interconexión hacia el nodo raíz mientras que el tráfico de entrada es distribuido desde el nodo raíz hacia las computadoras.

Como se puede ver en la Figura 3.6 es posible determinar la cantidad de tráfico que circula por medio de un determinado dispositivo de interconexión mediante la suma del tráfico generado por sus nodos hijos. Es decir:

$$C_t = \sum_1^n t_n \quad (3.4)$$

donde:

C_t es la cantidad de tráfico circulante total en el dispositivo de interconexión

n es el número de dispositivos hijo correspondientes a C

t_n es la carga de tráfico perteneciente a cada dispositivo hijo

Este dato es esencial para la construcción y funcionamiento de los modelos estadísticos dinámicos ya que éste determina la cantidad de tráfico que circula por algún dispositivo de interconexión en una determinada hora y minuto, sin embargo. Es necesario contar con un método capaz de almacenar la información de una topología de red en una base de datos, con el propósito de determinar la cantidad de tráfico circulante de cada dispositivo de interconexión. Debido a que una topología de árbol se encuentra formada en su totalidad por pares padre-hijo, una manera de almacenar dicha topología

Padre	Hijo
<i>nulo</i>	A
A	B
A	C
B	1
B	2
C	3
C	4

Cuadro 3.3: Pares padre-hijo de la Figura 3.6

es mediante una tabla que almacene todos los pares padre-hijo existentes en la topología de red utilizando un valor *nulo* para el padre del nodo raíz. Una vez obtenida dicha tabla es entonces posible conocer todas las diferentes rutas que pueden tomar los paquetes de red desde un punto en la red hasta otro punto de la misma. En el Cuadro 3.3 se muestra la tabla de pares padre-hijo de la topología mostrada en la Figura 3.6.

El número de pares padre-hijo de la tabla es igual al número de dispositivos de red contenidos en la topología, de esta manera es almacenada toda la topología de red así como todas las rutas de pares padre-hijo existentes.

Con la topología almacenada es posible determinar la ruta que siguen los paquetes de red durante su transmisión conociendo el punto de origen y destino de estos. La ruta se obtiene encontrando el conjunto de pares padre-hijo que conformen un camino desde el punto de origen hasta el punto destino de un paquete de red. La ruta obtenida permite calcular la carga de tráfico correspondiente a los distintos dispositivos de interconexión de la red. Por ejemplo la ruta de pares padre-hijo que sigue un paquete desde el nodo raíz “A” hasta la computadora “4” es: A-C→C-4, esto quiere decir que tanto “A” como “C” tienen una carga de tráfico igual a 1.

Al conocer las rutas usadas por cada paquete de red se tienen los datos necesarios para hacer uso de la fórmula anteriormente mencionada y es entonces posible determinar la carga de tráfico de cada uno de los dispositivos de interconexión la cual proporciona un dato esencial para la construcción de los modelos estadísticos dinámicos de cada dispositivo de interconexión ya que las estadísticas de dichos modelos se obtienen de la cantidad total de tráfico que circula por cierto dispositivo de interconexión en una hora y minuto determinado.

Con esto es posible la detección del conjunto de todos los dispositivos de interconexión anómalos, entendiendo como dispositivo anómalo al dispositivo cuyo tráfico

circulante presenta un comportamiento anómalo. Sin embargo, es necesaria la implementación de un método de rastreo de dispositivos anómalos ya que, debido al comportamiento jerárquico del tráfico de red en una topología de árbol, el comportamiento anómalo de un dispositivo de interconexión es heredado a sus nodos hijo o padre dependiendo de algunas características. Se rastrea el dispositivo de interconexión más cercano a la fuente de la anomalía considerando los modelos estadísticos de la ruta y la topología de red.

Para cumplir con ese propósito se dividen los dispositivos de interconexión anómalos en dos tipos: *dispositivos anómalos fuente* y *dispositivos anómalos víctima*. Un dispositivo anómalo fuente es aquel que ha ocasionado un comportamiento anormal en la red de área local debido a un mal funcionamiento propio u ocasionado por un comportamiento anómalo de sus computadoras hijas. Por otro lado, un dispositivo anómalo víctima es aquel dispositivo de interconexión cuyo comportamiento anómalo es ocasionado por la herencia de tráfico anómalo de alguno de sus dispositivos de interconexión padre o hijo.

Se aplican las siguientes reglas para hacer la clasificación de dispositivos anómalos:

- Un dispositivo anómalo es fuente cuando todos sus dispositivos de interconexión hijo presentan un comportamiento anómalo.
- Un dispositivo anómalo es fuente cuando sus dispositivos de interconexión hermanos presentan un comportamiento normal.
- Un dispositivo anómalo es víctima cuando uno o varios de sus dispositivos de interconexión hijo presentan un comportamiento normal.

Con el uso de estas reglas de rastreo, es posible separar el dispositivo anómalo fuente de los demás dispositivos de interconexión, tomando en cuenta los siguientes escenarios mostrados en la Figura 3.7.

Tipo A. El dispositivo de interconexión padre es identificado como dispositivo anómalo fuente ya que, todos sus dispositivos de interconexión hijos presentan un comportamiento anómalo.

Tipo B. Un dispositivo de interconexión hijo es identificado como dispositivo anómalo fuente ya que, sus demás dispositivos de interconexión hermanos presentan un comportamiento normal, dando como resultado que el dispositivo de interconexión padre sea identificado como el dispositivo anómalo víctima al heredar el tráfico anómalo.

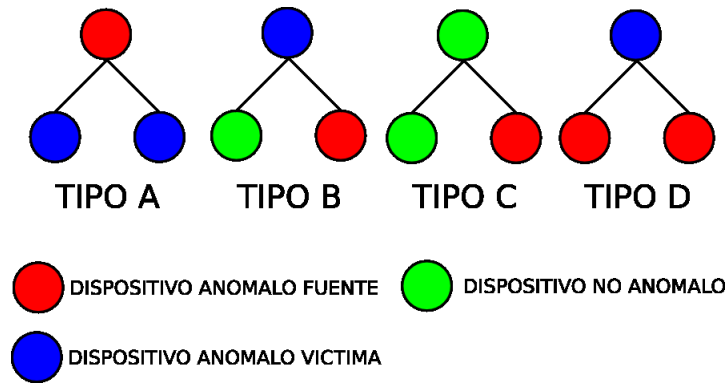


Figura 3.7: Escenarios de un comportamiento anómalo en dispositivos de interconexión.

Tipo C. No es necesario llevar a cabo un procedimiento de rastreo debido a que, solo existe un dispositivo de interconexión anómalo el cual no afecta el comportamiento de los demás dispositivos de interconexión. Este tipo de escenario es presentado cuando el tráfico circulante diario por dicho dispositivo es muy poco en comparación de los demás dispositivos, por lo cual una falla en éste no afecta las estadísticas de los demás dispositivos de interconexión.

Tipo D. El dispositivo de interconexión padre es el dispositivo anómalo víctima debido al comportamiento fallido de sus dispositivos de interconexión hijo, sin embargo, la probabilidad de que este escenario sea presentado en un caso real es remota, por tal motivo se omitió su uso para la generación de las reglas de rastreo antes mencionadas.

Al contar con modelos estadísticos dinámicos por cada dispositivo de interconexión de la red se obtiene un nivel más profundo y exacto del comportamiento del tráfico de la red de área local. Más aún al contar con la topología de la red y al aplicar las reglas de rastreo, es posible realizar la localización automática de un dispositivo de interconexión fallido específico o de una zona determinada afectada de la red, esto es de gran utilidad ya que en caso de detectar comportamientos anómalos en una zona de la red no es necesario detener por completo el servicio de ésta, sino tan solo el servicio de la zona o el dispositivo de interconexión anómalo. Lo anterior es explicado con la red de área local ejemplo mostrada en la Figura 3.8.

Esta red esta compuesta por tres dispositivos de interconexión: SWITCH, HUB1, HUB2, las cuales interconectan nueve computadoras. El comportamiento del tráfico de esta red fue simulado, provocando un ataque desde la computadora PC6 con una

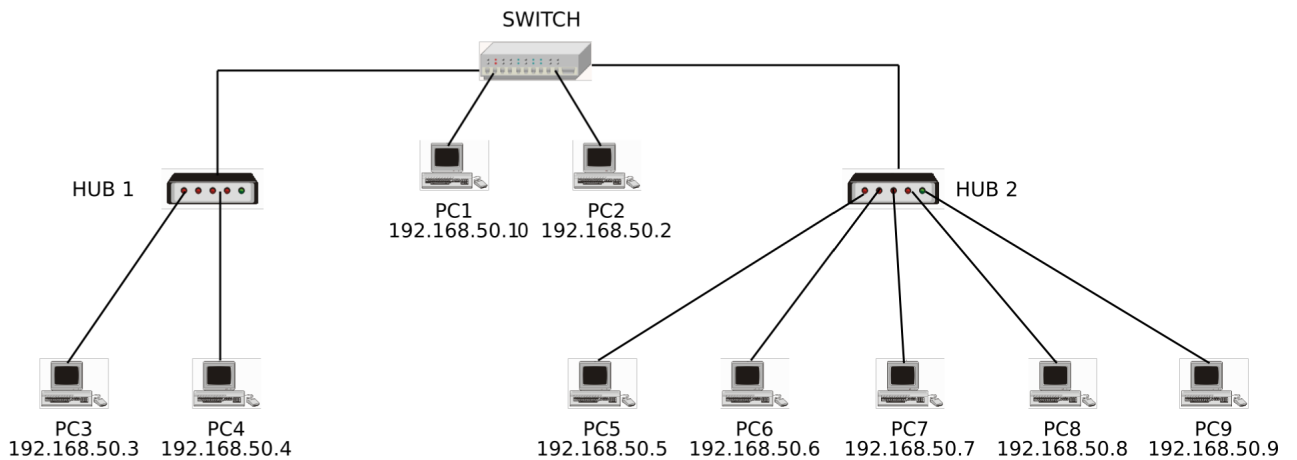


Figura 3.8: Red de área local.

dirección IP: 192.168.50.6 teniendo origen a las 10 horas y finalizando a las 14 horas del día. El resultado de este ataque fue un incremento inusual en el tráfico circulante de la red. En las Figuras 3.9, 3.10 y 3.11 se muestran las gráficas de los modelos estadísticos dinámicos correspondientes a los dispositivos de interconexión de esta red con el fin de mostrar su utilidad en conjunto con la topología de red.

En la Figura 3.9 se muestra la gráfica del modelo estadístico dinámico de HUB1. En esta se muestran tres comportamientos de tráfico diferentes, el primero de ellos es el tráfico real del dispositivo raíz, este muestra el tráfico total que circula a través del nodo raíz de la red de área local, en este caso se trata del tráfico circulante por SWITCH. El segundo comportamiento mostrado es el tráfico real del dispositivo actual, este muestra el comportamiento real del tráfico circulante a través del dispositivo de interconexión, en este caso se trata del tráfico circulante a través de HUB1, como se puede ver éste es solo una fracción del tráfico total de la red ya que corresponde al tráfico generado solamente por PC3 y PC4. El último comportamiento mostrado corresponde al modelo estadístico dinámico de HUB1, el cual sirve como medio de comparación entre el tráfico real y el esperado, como se puede ver el comportamiento real de este dispositivo no muestra señales de un ataque ya que es muy similar al comportamiento esperado por su modelo estadístico dinámico, esto se debe a que el ataque es originado por PC6 la cual es una máquina conectada a HUB2, por lo cual el tráfico de HUB1 no se ve afectado por éste.

En la Figura 3.10 se muestra la gráfica del modelo estadístico dinámico de HUB2.

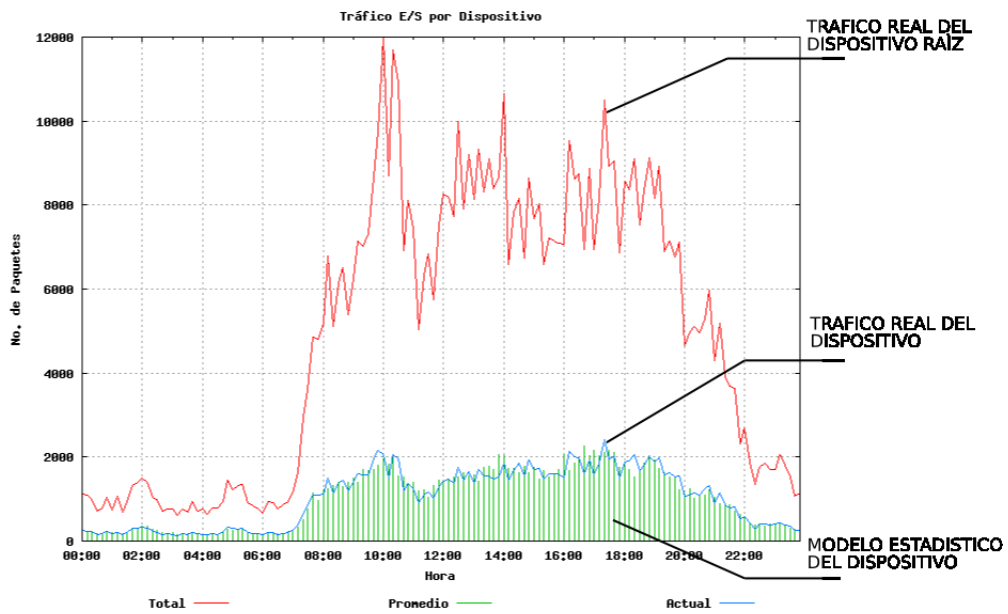


Figura 3.9: Modelo estadístico dinámico de HUB1

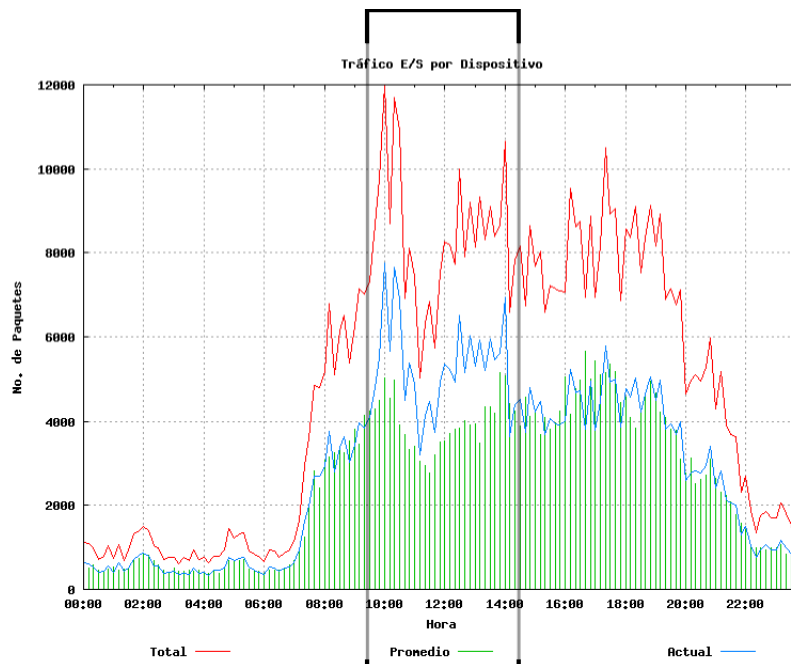


Figura 3.10: Modelo estadístico dinámico de HUB2

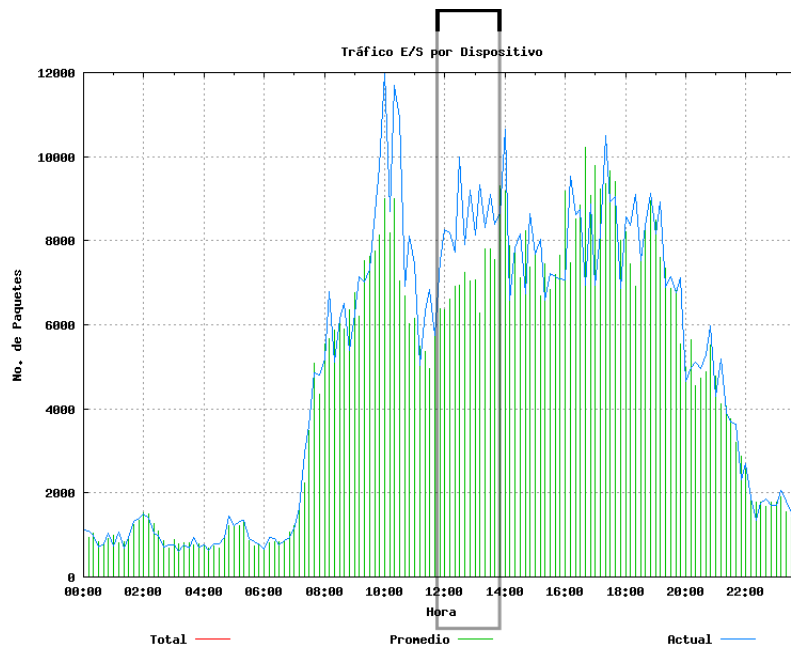


Figura 3.11: Modelo estadístico dinámico de SWITCH

En esta gráfica es posible observar los efectos del ataque originado por PC6. El tráfico anómalo corresponde a la zona delimitada por el rectángulo gris, como se puede observar en este caso su gráfica de comportamiento real se aleja de la gráfica correspondiente al modelo estadístico dinámico lo cual es una señal de la existencia de un comportamiento anómalo.

En la Figura 3.11 se muestra la gráfica del modelo estadístico dinámico de SWITCH. En esta sólo es posible ver dos comportamientos: el tráfico real del dispositivo y su modelo estadístico. No se muestra el comportamiento general de la red ya que, éste es el dispositivo raíz y ese comportamiento se duplicaría. A diferencia de HUB1, en este caso el comportamiento anómalo originado por PC6 tuvo repercusiones en el comportamiento de SWITCH como se muestra en el área delimitada por el rectángulo gris, como se puede ver el incremento de tráfico de HUB2 incrementó la carga de tráfico circulante por SWITCH, alejándolo del comportamiento esperado por su modelo estadístico.

En el Cuadro 3.4 se muestra la evolución del valor $\Delta e(d, h, c)$ para los dispositivos de interconexión de la red mostrada en la Figura 3.8, así como el resultado del método de rastreo.

Al usar los modelos estadísticos dinámicos en conjunto con la información de la topología de red es posible detectar comportamientos anómalos a nivel de dispositivo

Hora	$\Delta e(d, h, c) - \Delta$	$e(d, h, c) - \delta$	Comportamiento	Alarma	Dispositivos anómalos	Tipo	Fuente
9:50	$\Delta e(d, h, c) = 0\Delta t$	$e(d, h, c) < \delta$	Normal	Nula	-	-	-
10:00	$\Delta e(d, h, c) = 0\Delta t$	$e(d, h, c) > \delta$	Normal	Nula	-	-	-
10:10	$\Delta e(d, h, c) = 1\Delta t$	$e(d, h, c) > \delta$	Normal	Nula	HUB2	C	HUB2
10:20	$\Delta e(d, h, c) = 2\Delta t$	$e(d, h, c) > \delta$	Pico	Baja	HUB2	C	HUB2
10:30	$\Delta e(d, h, c) = 3\Delta t$	$e(d, h, c) > \delta$	Pico	Baja	HUB2	C	HUB2
10:40	$\Delta e(d, h, c) = 4\Delta t$	$e(d, h, c) > \delta$	Red	Media	HUB2	C	HUB2
10:50	$\Delta e(d, h, c) = 5\Delta t$	$e(d, h, c) > \delta$	Red	Media	HUB2	C	HUB2
11:00	$\Delta e(d, h, c) = 6\Delta t$	$e(d, h, c) > \delta$	Red	Media	HUB2	C	HUB2
⋮							
12:00	$\Delta e(d, h, c) = 12\Delta t$	$e(d, h, c) > \delta$	Ataque	Alta	SWITCH, HUB2	B	HUB2
⋮							
12:30	$\Delta e(d, h, c) = 15\Delta t$	$e(d, h, c) > \delta$	Ataque	Alta	SWITCH, HUB2	B	HUB2
⋮							
13:00	$\Delta e(d, h, c) = 18\Delta t$	$e(d, h, c) > \delta$	Ataque	Alta	SWITCH, HUB2	B	HUB2
⋮							
14:00	$\Delta e(d, h, c) = 24\Delta t$	$e(d, h, c) > \delta$	Ataque	Alta	SWITCH, HUB2	B	HUB2
14:10	$\Delta e(d, h, c) = 0\Delta t$	$e(d, h, c) < \delta$	Normal	Alta	-	-	-
14:20	$\Delta e(d, h, c) = 0\Delta t$	$e(d, h, c) < \delta$	Normal	Alta	-	-	-

Cuadro 3.4: Evolución del valor $\Delta e(d, h, c)$ y rastreo de dispositivos anómalos de la red mostrada en la Figura 3.8

de interconexión y no solo a nivel de red general, lo cual aumenta el nivel de detalle del tráfico de la red. Aunado a esto, usando el método de rastreo de dispositivos de interconexión anómalos, es posible detectar y separar el dispositivo anómalo fuente de los demás dispositivos de interconexión anómalos, con ello es posible conocer exactamente el dispositivo o la zona de la red fallida.

En resumen, las extensiones antes mencionadas permiten tener un nivel de visualización y de control de la red de área local más detallado ya que gracias a la incorporación de la información de la topología de red es posible no solo conocer la cantidad de tráfico que circula por el dispositivo raíz de la red sino también el tráfico que circula por cada uno de los dispositivos de interconexión. La actualización del modelo estadístico estático por un conjunto de modelos estadísticos dinámicos correspondientes a los distintos dispositivos de interconexión de la red ofrece la ventaja de contar con una guía siempre actualizada sobre el comportamiento esperado de la red de área local así como del comportamiento esperado de los distintos dispositivos de interconexión.

3.3. Detección automática de comportamientos anómalos

Un sistema de detección automática de comportamientos anómalos es aquel que con base en datos previos o reglas pre-establecidas es capaz de diferenciar un comportamiento anormal del tráfico de red de uno normal de manera automática para, de esta manera detectar posibles comportamientos anómalos que afecten el desempeño e integridad de la red. Estos sistemas pueden ser divididos por su fuente de información en detectores basados en red [8, 18, 25, 36] y detectores basados en máquinas [30, 33].

Los detectores basados en máquina fueron el primer tipo de detectores desarrollados e implementados. Su función es analizar los cambios a nivel de sistema operativo, es decir, controlar todos aquellos parámetros y archivos que están disponibles a nivel del sistema operativo, y que suelen ser modificados cuando tiene lugar un ataque. Estos sistemas actúan a nivel de máquina y requieren demasiado tiempo de gestión y configuración.

Los detectores basados en red funcionan analizando el tráfico de red, monitorizando así varias máquinas de la red. Un detector de este tipo bien localizado puede monitorizar una red grande, siempre y cuando tenga la capacidad suficiente para analizar todo el tráfico.

En la actualidad los sistemas basados en red más usados incluyen a NSM [20], Bro [30], NFR [25] y NetStat [15]. NSM es un sistema diseñado para monitorizar el tráfico entre los huéspedes en una red de área local. Bro funciona como un monitor de red pasivo, el cual filtra el tráfico de algunas aplicaciones. NFR fue diseñado como una herramienta flexible para la generación de informes de red. NestStat es un sistema detector de comportamientos anómalos que ofrece la personalización de colectores de eventos.

Existen sistemas que incluyen ambas opciones, monitorizando la red y las máquinas a la vez. Entre estos se encuentra Emerald [27], Grids [40] y Dids [38]. Emerald, fue diseñado para detectar intrusos en una red distribuida de gran tamaño. Grids acumula resultados de ambos, de componentes basados en máquinas y basados en red. Dids es una extensión de NSM, utiliza datos de archivos de auditoría de máquinas y análisis del tráfico de red para la detección de comportamientos anómalos.

Los sistemas anteriores pueden ser divididos en dos categorías más: basados en reglas o firmas y basados en anomalías. La detección basada en reglas es hecha por medio de un sistema experto filtrando la actividad de la red de acuerdo a un conjunto predefinido de reglas. La diferencia entre estos dos métodos reside en que los métodos basados en reglas no pueden detectar nuevos ataques para los cuales no tienen una regla definida, presentando un nivel de falsos positivos bajo. Por otro lado, los métodos de detección basados en anomalías tienen un rango más grande de falsos positivos, pero debido a que no necesitan reglas para hacer la detección, estos son capaces de detectar nuevos ataques.

Además de estos sistemas existen dos que son de licencia libre. Estos sistemas son Snort [36] y Shadow [39]. Snort es un sistema basado en reglas para redes relativamente pequeñas. Shadow recae fuertemente en la utilización de tcpdump [43].

El sistema desarrollado es un sistema detector de comportamientos anómalos basado en red ya que recibe los datos del tráfico por medio del sistema monitor de tráfico. Por su método de análisis está clasificado como un sistema basado en anomalías, debido a que la búsqueda de comportamientos anómalos se efectúa por medio de los modelos estadísticos dinámicos de cada dispositivo de interconexión sin embargo, como veremos más adelante no todos los comportamientos anómalos son susceptibles de ser detectados mediante métodos basados en anomalías, por tal motivo se desarrolló un modulo capaz de efectuar una búsqueda de estos.

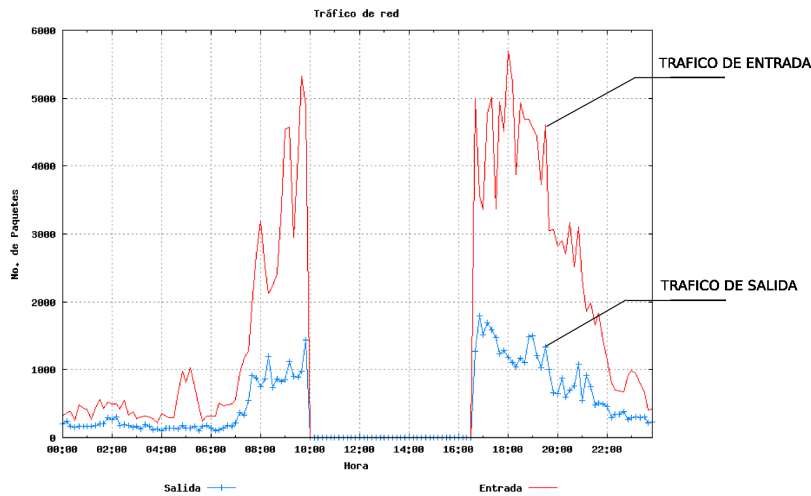


Figura 3.12: Tráfico de red durante una falla

3.3.1. Detección basada en anomalías

En esta parte se muestran los comportamientos anómalos que son capaces de ser detectados por medio de los modelos estadísticos dinámicos así como el comportamiento característico de cada uno de ellos.

Falla

Una falla puede entenderse como una deficiencia que se presenta de manera natural en el funcionamiento de una red de área local. Esto puede ocurrir cuando por ejemplo, debido al mal uso o el deterioro sufrido a través del tiempo, los cables de la red no transfieren de manera normal la información, ocasionando un bloqueo en la circulación del tráfico. Otro tipo de falla en la que una red puede verse afectada es el desperfecto de un concentrador, conmutador, o servidor. Esto sucede cuando un dispositivo deja de transmitir información, lo cual se puede atribuir a diferentes causas, desde la falta de energía eléctrica hasta la saturación de tráfico en el dispositivo.

El efecto que causa una falla en el comportamiento de la red es una interrupción total del tráfico circulante, ya que tanto el tráfico de entrada como el de salida son nulos durante la presencia de la falla. En la Figura 3.12 se muestra un gráfico en la que se muestra el comportamiento de la red durante una falla.

En esta gráfica se muestra el número de paquetes de entrada y salida circulante en la red durante todo un día de labores. En este caso en particular se muestra una falla

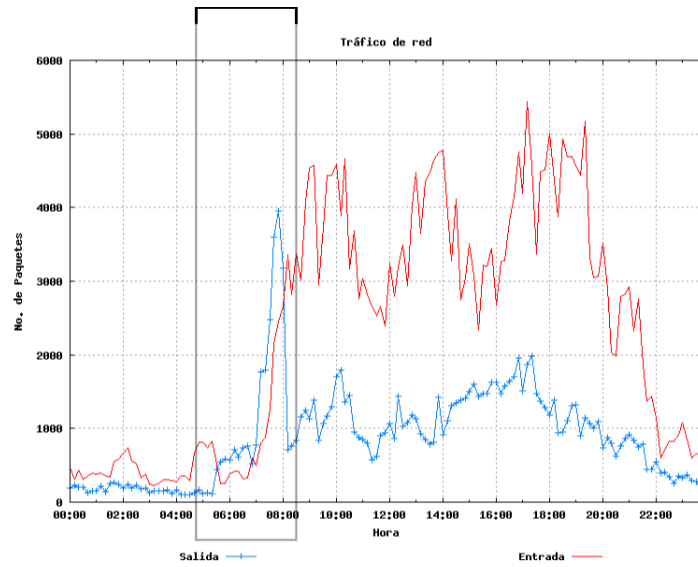


Figura 3.13: Tráfico de red durante un abuso

presentada a partir de las 10:00 horas hasta las 16:30 horas, donde se puede ver que ambos tipos de tráfico son completamente nulos en este lapso de tiempo, normalizándose posteriormente.

Este tipo de comportamientos anómalos es fácil de detectar mediante el uso de los modelos estadísticos ya que debido a la presencia del tráfico nulo, el comportamiento de la red se aleja demasiado del comportamiento esperado por el modelo estadístico haciendo posible su detección.

Abusos

Un abuso en una red, a diferencia de las fallas, ocurre cuando de manera predeterminada un usuario hace un mal uso de los recursos de la red. Un ejemplo de un abuso es la saturación de recursos. Este tipo de abuso se ocasiona cuando se hace circular por la red una gran cantidad de tráfico. Este tipo de información puede ser causada por servidores multimedia no autorizados, los cuales ponen a disposición archivos de audio o vídeo, saturando de esta manera los servidores de la red. Otro tipo de abuso se puede presentar en la dirección opuesta, es decir, usuarios externos que saturan de información ajena la red local. Este comportamiento puede ser detectado cuando se presenta por períodos prolongados lo cual, los hace claramente visibles al compararlos con el modelo del comportamiento esperado. En la Figura 3.13 se muestra el tráfico de

entrada y salida de una red de área local, en el cual se presenta un abuso interno, en el lapso de las 5:30 a 8:00 horas. Como se puede ver dentro de este lapso de tiempo la cantidad de paquetes de salida sobrepasan incluso la cantidad de paquetes de entrada.

Este tipo de comportamiento en menor magnitud puede presentarse en el sentido contrario, es decir un incremento en el tráfico de entrada, debido a un usuario descargando una gran cantidad de información. Normalmente este tipo de abuso pasa desapercibido en sistemas detectores que sólo supervisan el tráfico general de la red sin embargo, en nuestro caso pueden ser detectados a nivel de dispositivo de interconexión gracias a los modelos estadísticos dinámicos.

Ataques

Un ataque se presenta cuando usuarios ajenos a la red buscan alterar el desempeño de la misma. Debido a que estos usuarios no cuentan con acceso legítimo a la red, la única manera de lograr su objetivo es explotando las vulnerabilidades existentes.

Un ataque de negación de servicio es un ejemplo de este tipo de comportamiento, en éste, el atacante ha conseguido encontrar un puerto o dirección IP vulnerable. Utilizando esta vulnerabilidad el atacante intenta sobrecargar la capacidad de manejo de paquetes de red de algún servidor. El resultado será un incremento anormal en el número de paquetes de entrada que circulan en la red. En la mayoría de los casos los ataques se caracterizan por un comportamiento inusual en el tráfico de entrada, ya que es el único al cual pueden tener acceso.

En la Figura 3.14 se muestra el comportamiento de una red de área local durante un ataque de negación de servicio. Como se puede observar el número de paquetes de entrada aumenta significativamente entre las 10:00 y 16:00 horas volviendo a la normalidad posteriormente. Este tipo de ataque es fácil de detectar mediante los modelos estadísticos dinámicos debido a la gran cantidad de tráfico que genera el ataque en comparación con un comportamiento normal.

Gusanos

Los gusanos son programas que se reproducen de una computadora a otra ya que, cuentan con la habilidad de viajar y reproducirse sin la ayuda de una persona o intervención alguna. Un solo gusano puede generar y enviar cientos o miles de copias de sí mismo a través de una red, causando un efecto devastador en la calidad de servicio de la misma consumiendo demasiado o todo el ancho de banda de esta.

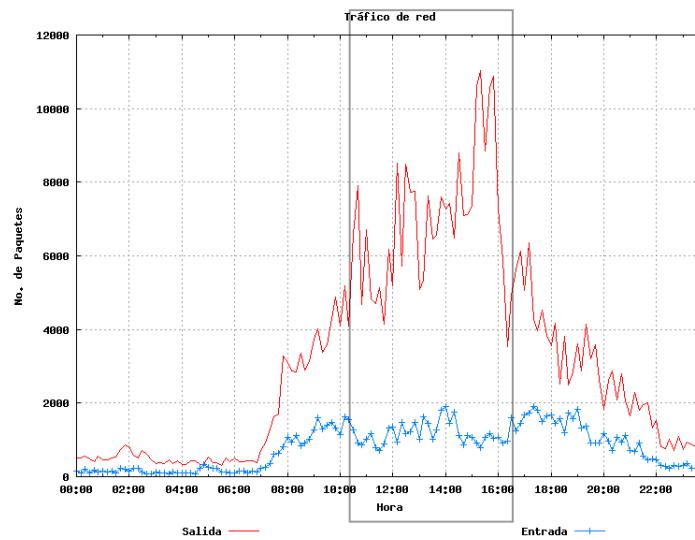


Figura 3.14: Tráfico de red durante un ataque

	Tráfico	
	Entrada	Salida
Falla	Nulo	Nulo
Abuso	Excesivo / Normal	Excesivo / Normal
Ataque	Excesivo	Normal
Gusano	Normal	Excesivo

Cuadro 3.5: Características de comportamientos anómalos

En la Figura 3.15 se muestra el comportamiento de una red de área local durante la proliferación de un gusano, cuyos efectos se pueden ver entre las 14:10 y 20:00 horas, donde la cantidad de paquetes de salida aumentó significativamente en comparación con el resto del día.

Una característica del efecto causado por un gusano es que siempre utilizará el tráfico de salida de la red ya que su misión es hacer el mayor número de copias de sí mismo hasta agotar los recursos de la red.

En el Cuadro 3.5 se muestran los comportamientos característicos de los comportamientos antes mencionados.

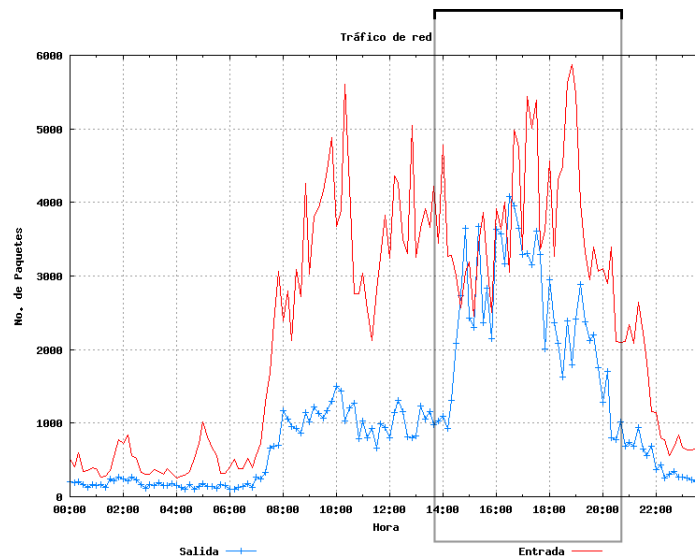


Figura 3.15: Tráfico de red durante la proliferación de un gusano

3.3.2. Comportamientos anómalos basados en reglas

Existen casos donde un comportamiento anómalo no afecta de manera significativa el tráfico de la red por lo cual éste no puede ser detectado usando métodos estadísticos. Debido a esto, es necesario agregar módulos detectores de intentos de intrusión como un refuerzo para incrementar el nivel de detección de comportamientos anómalos del sistema.

Para que un intruso obtenga acceso a la red, debe primero encontrar puntos vulnerables dentro de la misma. Existen dos técnicas que son las más usadas para encontrar puntos de acceso o afectar el desempeño de la red para obtener algún beneficio propio.

Estas dos técnicas son la *exploración de puertos* y la *exploración de direcciones IP*. Para poder comprender cómo funcionan la exploración de puertos y direcciones IP, es imprescindible conocer ciertos aspectos técnicos sobre el protocolo TCP/IP. En la arquitectura TCP/IP, las conexiones se realizan mediante el establecimiento de *sockets*. Un socket es la combinación de un par que contiene el número IP de una máquina y un número de *puerto* con otro par, el cual contiene el número IP de otra o la misma máquina y otro número de puerto.

Toda conexión en redes TCP/IP están identificadas por un socket, es decir, una máquina de origen con su puerto, y una máquina destino con su puerto; cada máquina es identificada por su número IP.

Clave	Clave / Valor	Clave / Valor	Valor
IP fuente	IP destino	puerto fuente	puerto destino

Cuadro 3.6: Estructura de la tabla hash usada para la exploración de puertos

Hora	IP Fuente	IP Destino	Puerto Fuente	Puerto Destino	Δp	Estado
10:50	148.201.50.25	192.168.50.1	3265	2020	$\Delta p = 0$	Normal
10:50	148.201.50.25	192.168.50.1	3265	2021	$\Delta p = 1$	Normal
10:50	148.201.50.25	192.168.50.1	3265	2022	$\Delta p = 2$	Normal
10:50	148.201.50.25	192.168.50.1	3265	\vdots	\vdots	
10:50	148.201.50.25	192.168.50.1	3265	2132	$\Delta p = 112$	Exploración
10:50	148.201.50.25	192.168.50.1	3265	2133	$\Delta p = 113$	Exploración

Cuadro 3.7: Comportamiento de los pares IP+puerto y Δp durante una exploración de puertos

3.3.2.1. Exploración de puertos

La exploración de puertos es una técnica usada para descubrir canales de comunicación aprovechables en un servidor. Esta técnica consiste en realizar una exploración o barrido de puertos de uno o un grupo de máquinas con la finalidad de obtener un listado de los puertos abiertos del sistema.

Cuando una sesión normal de red es iniciada desde una máquina es normal el uso de cierta cantidad de puertos, ya sea porque se use el puerto 80 para navegar por la Web, el puerto 25 para enviar correos electrónicos, entre otros. Sin embargo, estos números de puerto son difícilmente secuenciales. Por otro lado, cuando una exploración de puertos es efectuada, se presenta un patrón con un gran número de puertos destino usados de manera secuencial.

Para efectuar la detección de exploraciones de puertos se hace uso de la variable Δp en conjunto de tablas hash para su análisis. Una tabla hash es un contenedor asociativo que permite un almacenamiento de elementos (denominados valores) a partir de otros objetos (llamados claves) [2]. La estructura de la tabla hash usada es mostrada en el Cuadro 3.6.

El objetivo de dicho análisis es la detección de las tablas hash que rebasen el límite establecido por Δp . El valor usado para la detección de exploraciones de puertos es de $\Delta p > 50$ en un solo lapso de tiempo Δt . Un ejemplo es mostrado en el Cuadro 3.7, en el cual se muestra una exploración de puertos y la evolución de la variable Δp durante

Clave	Clave / Valor	Clave / Valor	Valor
IP fuente	puerto fuente	puerto destino	IP destino

Cuadro 3.8: Estructura de la tabla hash usada para la exploración de direcciones IP

IP Fuente	Puerto Fuente	Puerto Destino	IP Destino	Δd	Estado
148.201.50.223	1523	4132	192.168.50.20	$\Delta d = 0$	Normal
148.201.50.223	1523	4132	192.168.50.21	$\Delta d = 1$	Normal
148.201.50.223	1523	4132	192.168.50.22	$\Delta d = 2$	Normal
148.201.50.223	1523	4132	\vdots	\vdots	
148.201.50.223	1523	4132	192.168.50.60	$\Delta d = 40$	Exploración
148.201.50.223	1523	4132	192.168.50.62	$\Delta d = 41$	Exploración
148.201.50.223	1523	4132	192.168.50.63	$\Delta d = 42$	Exploración

Cuadro 3.9: Comportamiento de los pares IP+puerto y Δd durante una exploración de direcciones IP

la misma.

El algoritmo usado para la detección de exploraciones de puerto es explicado con más detalle en el siguiente capítulo.

3.3.2.2. Exploración de direcciones IP

Otra manera en la que se trata de burlar la seguridad de la red es aquella en la cual el atacante se hace pasar por un usuario legítimo de la misma. Esto se logra obteniendo una dirección IP de la red local, con lo cual el atacante puede ocultar su identidad mostrándose como un usuario legítimo.

De la misma manera en la que una exploración de puertos presenta un patrón generalizado sin importar que técnica o herramienta sea usada. La exploración de direcciones IP también presenta un patrón en particular. Cuando una exploración de direcciones IP es efectuado los paquetes fluyen a través de una misma dirección IP y puerto fuente hacia un mismo número de puerto destino de varias direcciones IP consecutivas.

Para efectuar la búsqueda de este tipo de exploraciones se hace uso de la variable Δd en conjunto de la tabla hash cuya estructura es mostrada en el Cuadro 3.6. El valor límite de Δd usada para la detección de exploraciones es de $\Delta d > 30$. Un ejemplo de una exploración de direcciones IP y de la evolución del valor Δd es mostrado en el Cuadro 3.9, como se puede ver las direcciones IP destino forman un patrón consecutivo.

El algoritmo usado para la detección de exploraciones de direcciones IP es explicado con más detalle en el siguiente capítulo.

Capítulo 4

Supervisor de red

Con las extensiones desarrolladas para el sistema monitor de red, es posible efectuar la detección automática de comportamientos anómalos en una red de área local. Mediante el uso de los modelos estadísticos dinámicos se efectúa la detección de comportamientos anómalos en los dispositivos de interconexión de la red de área local. Añadiendo a esto el monitoreo de exploraciones de puertos y direcciones IP es posible detectar intentos de intrusión en el tráfico de red.

En este capítulo se muestra la arquitectura del sistema desarrollado, mostrando los distintos módulos que lo conforman y la forma en que interactúan mutuamente.

En la Figura 4.1 se muestra la arquitectura general del sistema desarrollado. El sistema se encuentra dividido por su funcionalidad en dos fases, la primera de ellas utiliza los datos proporcionados por el módulo simulador de tráfico de red con el fin de realizar pruebas de desempeño en la etapa de desarrollo. En la segunda fase el sistema recibe los datos del tráfico de red real proporcionados por el sistema monitor de red.

Los módulos del sistema inician su función cada lapso de tiempo Δt ($t = 10$ min.) en el siguiente orden:

Simulación de tráfico de red. Módulo encargado de generar tráfico artificial para la etapa de desarrollo y pruebas.

Detección de fallas y abusos. Módulo encargado de efectuar la comparación entre el comportamiento real y el esperado por los modelos estadísticos de los dispositivos de interconexión en busca de comportamientos anómalos.

Rastreo de dispositivos anómalos. Localiza de entre todos los dispositivos anómalos encontrados al dispositivo de interconexión más cercano a la fuente del com-

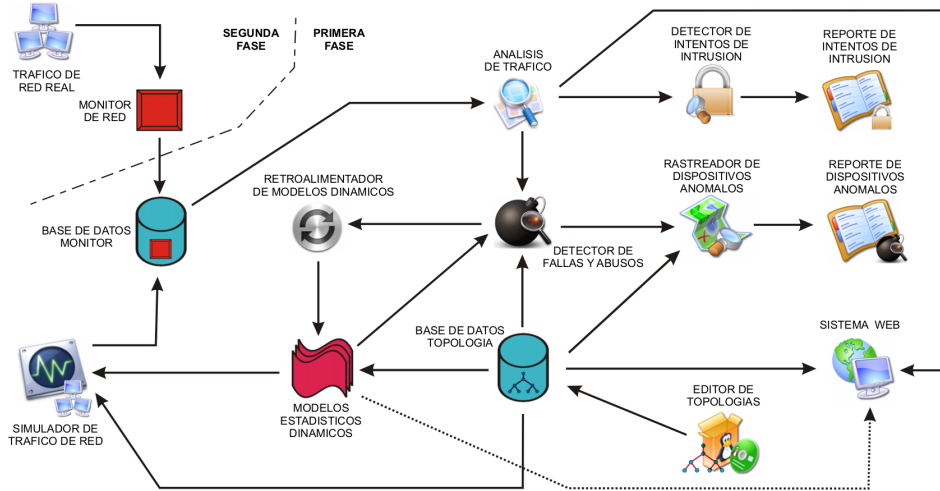


Figura 4.1: Arquitectura general del sistema de supervisión de redes de área local

portamiento anómalo.

Detección de intentos de intrusión. Módulo encargado de detectar intentos de intrusión mediante la búsqueda de patrones de exploraciones de red en el tráfico.

Actualización de modelos estadísticos. En este módulo se actualiza el comportamiento esperado de los dispositivos de interconexión cuyo comportamiento fue encontrado libre de anomalías.

La etapa de simulación es eliminada en la segunda fase del sistema para recibir los datos del tráfico directamente desde el monitor de red. En las secciones siguientes se explicará con detalle la funcionalidad de cada uno de los módulos mostrados.

4.1. Base de datos

El sistema hace uso de una base de datos la cual es la encargada de almacenar toda la información necesaria para el funcionamiento del sistema supervisor de red. Para la creación y manejo de la base de datos se utilizó el administrador de base de datos *mysql*. La base de datos sólo puede ser accedida por un usuario autorizado el cual debe ser autenticado mediante una clave de acceso.

El esquema general de la base de datos se muestra en la Figura 4.2. La base de datos esta dividida en dos partes, la primera parte esta constituida por las tablas *gra24*, *datos*

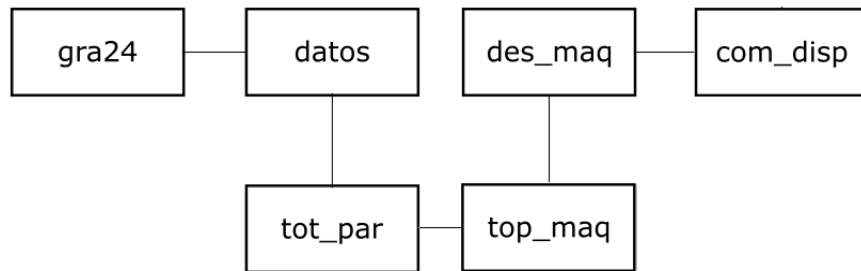


Figura 4.2: Tablas de la base de datos.

Campo	Tipo	Descripción
dia	varchar (4)	Fecha de la medición (mmdd)
hora	varchar (4)	Hora de la medición (hhmm)
proto	varchar (50)	Protocolo usado
fuelle	int unsigned (10)	IP fuente
destino	int unsigned (10)	IP destino
pfuelle	smallint unsigned (5)	Puerto fuente
pdestino	smallint unsigned (5)	Puerto destino

Cuadro 4.1: Campos correspondientes a la tabla *datos*

y *tot_par*, las cuales almacenan los datos del tráfico de red. La segunda parte pertenece a la topología de red almacenada en las tablas *des_maq*, *top_maq* y *com_disp*.

A continuación se describe la función que desempeña cada una de las tablas.

datos. En esta tabla se guarda la información de cada uno de los paquetes que circula por la red de área local. En el Cuadro 4.1 se pueden ver los campos que conforman esta tabla.

tot_par. En esta tabla son guardadas las estadísticas de los paquetes transmitidos por la red. A diferencia de la tabla *datos* en esta tabla se guarda la cantidad de paquetes que entran y salen por cada dirección IP de la red. Los campos de esta tabla son mostrados en el Cuadro 4.2.

gra24. En esta tabla se guardan los datos necesarios para la realización de una gráfica correspondiente al comportamiento total del tráfico de red. Los campos que con-

Campo	Tipo	Descripción
dia	varchar (4)	Fecha de la medición (mmdd)
hora	varchar (4)	Hora de la medición (hhmm)
total	varchar (20)	Total de paquetes transmitidos
host	int (11)	Dirección IP
entran	int (11)	Paquetes de entrada
salen	int (11)	Paquetes de salida
tcp	int (11)	Paquetes TCP
udp	int (11)	Paquetes UDP
icmp	int (11)	Paquetes ICMP
otro	int (11)	Paquetes de otros protocolos

Cuadro 4.2: Campos correspondientes a la tabla *tot_par*

Campo	Tipo	Descripción
dia	varchar (4)	Fecha de la medición (mmdd)
hora	varchar (4)	Hora de la medición (hhmm)
entran	int (11)	Total de paquetes de entrada
salen	int (11)	Total de paquetes de salida
modificado	smallint (1)	Bandera

Cuadro 4.3: Campos correspondientes a la tabla *gra24*

tiene esta tabla son mostrados en el Cuadro 4.3.

des_maq. La función que tiene esta tabla es la de guardar los datos que describen a cada uno de los dispositivos que conforman la red de área local. Los campos que componen esta tabla son mostrados en el Cuadro 4.4. Cada dispositivo es identificado con la relación mostrada en el Cuadro 4.5

top_maq. La función de esta tabla es la de almacenar la información de la topología de la red. La tabla consta de cuatro campos los cuales son mostrados en el Cuadro 4.6.

com_disp. En esta tabla son almacenados los modelos estadísticos dinámicos de cada dispositivo de interconexión. En el Cuadro 4.7 se muestran los diferentes cam-

Campo	Tipo	Descripción
tipo	int (11)	Tipo de dispositivo
ip	char (15)	Dirección IP o nombre
nombre	char (50)	Nombre del dispositivo
num_puertos	int (11)	Número de puertos
con_puerto	int (11)	Puerto de conexión
descripcion	char (100)	Descripción del dispositivo

Cuadro 4.4: Campos correspondientes a la tabla *des_maq*

Clave	Tipo
1	Concentrador
2	Conmutador
3	Puerta de enlace
4	Computadoras

Cuadro 4.5: Claves numéricas correspondientes al campo *tipo* perteneciente a la tabla *des_maq*

Campo	Tipo	Descripción
tipo_p	int (11)	Tipo del dispositivo padre
ip_padre	char (15)	IP o nombre del dispositivo padre
tipo_h	int (11)	Tipo del dispositivo hijo
ip_hijo	char (15)	IP o nombre del dispositivo hijo

Cuadro 4.6: Campos correspondientes a la tabla *top_maq*

Campo	Tipo	Descripción
dispositivo	char (20)	IP o nombre del dispositivo
num_día	int (11)	Día de comportamiento
hora	char (4)	Hora del comportamiento (hhmm)
comportamiento	int (11)	Paquetes esperados
actual	int (11)	Paquetes circulantes

Cuadro 4.7: Campos correspondientes a la tabla *com_disp*

Clave	Día
0	Domingo
1	Lunes
2	Martes
3	Miércoles
4	Jueves
5	Viernes
6	Sábado

Cuadro 4.8: Claves numéricas correspondientes al campo *num_día* de la tabla *com_disp*

pos que componen a esta tabla. Los días son almacenados mediante las claves numéricas mostradas en el Cuadro 4.8.

4.2. Detección de fallas y abusos

La detección de fallas y abusos es realizada utilizando la fórmula 3.2 en conjunto con los datos contenidos en la tabla *com_disp*, utilizando los campos *comportamiento* y *real* para las variables $f_m(d, h, c)$ y $f_r(d, h, c)$ respectivamente, donde $f_m(d, h, c)$ es el número de paquetes circulantes esperados por el modelo y $f_r(d, h, c)$ es el número de paquetes reales. Dicha comparación tiene como objetivo determinar el valor del error esperado $e(d, h, c)$ para determinar la diferencia entre el comportamiento real y el esperado por los modelos estadísticos dinámicos. La magnitud de dicha diferencia es medida usando la fórmula ?? para determinar si su valor rebasa el límite establecido por la frontera de decisión δ . Esta medida es usada para la detección y filtrado de los dispositivos de interconexión que presentaron un comportamiento anómalo, y es posible actualizar los modelos dinámicos de los dispositivos de interconexión que presentaron un comportamiento normal. En el algoritmo 1 se muestra el método utilizado.

Algoritmo 1 Detección de fallas y abusos en dispositivos de interconexión

ENTRADA: Datos de la tabla *tot_par*, datos de la tabla *top_maq*, e información contenida en el campo *comportamiento* de la tabla *com_disp*

SALIDA: Lista de dispositivos anómalos y lista de dispositivos normales

1. Por cada dispositivo
 - a) Obtener el valor $f_r(d, h, c)$
 - b) Obtener el valor $f_m(d, h, c)$
 - c) Si $e(d, h, c) > \delta \wedge \Delta e(d, h, c) > \Delta t$
 - 1) Agregar dispositivo a la lista de dispositivos anómalos
 - d) Si $e(d, h, c) \leq \delta$
 - 1) Agregar dispositivo a la lista dispositivos normales
-

Al finalizar su ejecución, este módulo obtiene dos listas, la primera de ella es la lista de dispositivos anómalos la cual es utilizada por el módulo de rastreo de dispositivos anómalos. La segunda lista contiene los dispositivos cuyo comportamiento fue considerado normal y es utilizada por el módulo de actualización de modelos estadísticos.

4.3. Rastreo de dispositivos anómalos

El módulo rastreador de dispositivos anómalos tiene como objetivo identificar y separar los dispositivos anómalos fuente de los dispositivos anómalos víctima, esto, con el propósito de detectar el dispositivo de interconexión más cercano a la fuente del comportamiento anómalo. El rastreo es efectuado usando las reglas de rastreo de dispositivos anómalos fuente establecidas en la sección 3.2.2, usando la información de la tabla *top_maq*.

Para realizar el rastreo automático se hace uso del algoritmo 2, ofreciendo como resultado final un análisis más exacto de la fuente del comportamiento anómalo.

El resultado entregado consiste una lista que contiene a los dispositivos anómalos fuente. Esta lista es utilizada para generar los reportes de fallas y abusos del sistema.

Algoritmo 2 Rastreo automático de dispositivos anómalos fuente

ENTRADA: Lista de dispositivos anómalos y datos contenidos en la tabla *top_maq*

SALIDA: Lista de dispositivos anómalos fuente

1. Para cada dispositivo anómalo en la lista
 - a) Encontrar dispositivos de interconexión hijos
 - b) Para cada dispositivo hijo
 - 1) Buscar dentro de la lista
 - c) Si todos los dispositivos hijos se encuentran en la lista
 - 1) Borrar a los dispositivos hijos de la lista
 - d) Si no todos los dispositivos hijos están dentro de la lista
 - 1) Borrar al dispositivo padre de la lista
-

4.4. Actualización de modelos estadísticos

Este módulo tiene como objetivo realizar la actualización continua de la información de los modelos estadísticos almacenados en la tabla *com_disp*, utilizando los valores contenidos en los campos *comportamiento* y *actual* para obtener el valor de las variables $f_m(d, h, c)$ y $f_r(d, h, c)$ respectivamente, donde $f_m(d, h, c)$ es el valor del modelo almacenado en el campo *comportamiento* y $f_r(d, h, c)$ es el número de paquetes reales. El proceso de retroalimentación se obtiene reemplazando el valor $f_m(d, h, c)$ por el valor obtenido $f_{new}(d, h, c)$, el cual es nuevo valor del modelo como se muestra en la fórmula 3.3.

En el algoritmo 3 se muestra la metodología usada para la retroalimentación de los modelos estadísticos dinámicos.

4.5. Detección de intentos de intrusión

La detección de intentos de intrusión se lleva a cabo mediante la búsqueda de patrones de exploración de red; exploraciones de puertos y de direcciones IP específicamente. Para realizar dicha detección, este módulo utiliza los datos contenidos en la tabla *datos* utilizando los campos *fuelle*, *destino*, *pfuelle* y *pdestino*. El procedimiento de detección de exploraciones de red es presentado en el algoritmo 4, usando las variables Δp y Δd explicadas en la sección 3.3.2. El algoritmo muestra el uso de las cuádr-

Algoritmo 3 Retroalimentación de los modelos estadísticos dinámicos

ENTRADA: Lista de dispositivos normales y comportamiento real de cada dispositivo

SALIDA: Actualización del comportamiento esperado de los dispositivos de interconexión

1. Para cada dispositivo
 - a) Si no existe registro
 - 1) Insertar valor $f_r(d, h, c)$
 - b) Si existe el registro
 - 1) Si $f_m(d, h, c) < f_r(d, h, c)$
 $a' m = 1$
 - 2) Otro, si $f_m(d, h, c) = f_r(d, h, c)$
 $a' m = 0$
 - 3) Otro, si $f_m(d, h, c) > f_r(d, h, c)$
 $a' m = -1$
 - 4) $f_{new}(d, h, c) = f_m(d, h, c) + ((f_r(d, h, c) * g) * m)$
 - 5) Insertar valor $f_{new}(d, h, c)$
-

Algoritmo 4 Detección de exploraciones de red

ENTRADA: Información de la tabla *datos* enviada por el módulo de análisis del tráfico de red.

SALIDA: Listas de cuádruplas sospechosas de presentar un patrón de comportamiento correspondiente a un exploración de red.

1. Para cada dirección IP fuente
 - a) Almacenar cuádruplas IP fuente, puerto fuente, IP destino, puerto destino
 - b) Almacenar cuádruplas IP fuente, puerto fuente, puerto destino, IP destino
 2. Detectar cuádruplas con un patrón correspondiente a un exploración de puertos
 - a) Para cada tripleta ip_fuente-ip_destino-puerto_fuente iguales
 - 1) num_puerto_destino++
 - b) Si num_puerto_destino $> \Delta p$
 - 1) Clasificar cuádruplas como anómalas
 3. Detectar cuádruplas con un patrón correspondiente a un exploración de direcciones IP
 - a) Para cada tripleta ip_fuente-puerto_fuente-puerto_destino iguales
 - 1) num_ip_destino++
 - b) Si num_ip_destino $> \Delta d$
 - 1) Clasificar cuádruplas como anómalas
-

plas almacenadas en tablas hash con el fin de realizar la comparación de las mismas usando las variables Δp y Δd para detectar exploraciones de puertos y direcciones IP respectivamente.

4.6. Simulador de tráfico de una red de área local

Existen varios factores que motivaron el desarrollo de un simulador de red, uno de ellos es la ventaja de contar con tráfico de red realista sin la necesidad de utilizar la red real, con lo cual es posible contar con los datos del tráfico de red correspondiente a un día entero de comportamiento en tan solo unos instantes.

La mayoría de los simuladores de red actuales recaen en una de las siguientes dos

categorías [23]:

Simuladores fuera de línea. Son usualmente herramientas altamente configurables y extendibles, diseñadas para simular procesos de una red en una escala de tiempo virtual la cual, no está linealmente relacionada con el tiempo real. Las simulaciones son ejecutadas dependiendo de tareas candelarizadas cuyo resultado puede ser analizado después de completada la simulación.

Simuladores de tiempo real. Son herramientas capaces de crear topologías de red virtuales y simular los efectos del tráfico en tiempo real o escalado. Dichos simuladores son generalmente menos flexibles que su contraparte fuera de línea; sin embargo, su mayor ventaja reside en su capacidad de análisis de resultados durante la simulación.

El simulador desarrollado pertenece a la categoría de simuladores en tiempo real ya que éste reproduce al comportamiento de una red de área local en una escala de tiempo lineal al tiempo real, además de reproducir dicho comportamiento dependiendo de una topología de red propuesta.

La simulación de tráfico es obtenida mediante la reproducción artificial del comportamiento de las computadoras registradas en la base de datos de la topología de red. Usando dichos datos en conjunto con la información proporcionada por los modelos estadísticos es posible obtener un comportamiento de tráfico de red realista.

Para que la información contenida en los modelos estadísticos sea la apropiada para realizar diferentes sesiones de simulación con un comportamiento de red diferente en cada una, es necesario llevar a cabo una alteración de los mismos, mediante la cual es posible obtener datos de tráfico correspondientes a un comportamiento normal de red, así como fallas y abusos de la misma.

Cada etapa del simulador es ejecutada en lapsos de 5 segundos en el siguiente orden: Alteración del modelo, generación de tráfico artificial, generación de exploraciones de red, generación de fallas y abusos. En dichas etapas son simulados 10 minutos de tráfico de red teniendo un total de 144 ejecuciones en una sola simulación. Los resultados de la simulación y el análisis pueden ser observados en tiempo real mediante la interfaz Web.

En las siguientes secciones son mostradas las etapas del simulador de tráfico de red desarrollado cuya arquitectura es mostrada en la Figura 4.3.

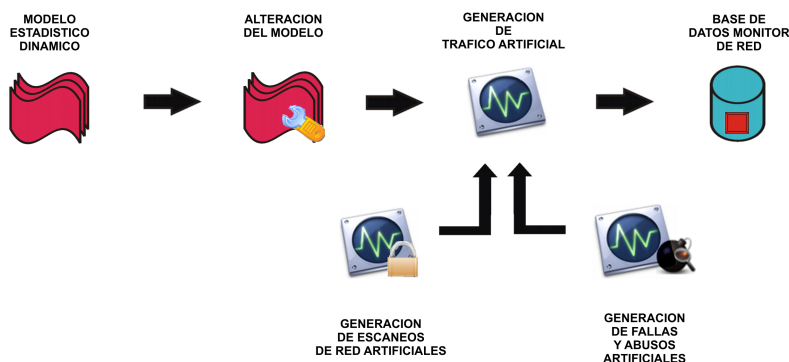


Figura 4.3: Arquitectura del simulador de tráfico de red

4.6.1. Alteración del modelo estadístico dinámico

El objetivo de efectuar una alteración del modelo estadístico es el de obtener la aleatoriedad presentada por un comportamiento de tráfico de red real. Para llevar a cabo dicha operación es utilizada la información de la tabla *com_disp*. La alteración de los modelos estadísticos es obtenida mediante el uso de la siguiente formula:

$$\begin{aligned}
 M &= \{x_k, y_k\}, k = 0, \dots, n - 1 \\
 P &= \{x_k, y_k^p\}, y_k^p = y_k + m_k \\
 m_k &= r_k \times f \times y_k \\
 r_k &\in [-1, 1]
 \end{aligned}
 \tag{4.1}$$

donde:

M son los datos del modelo estadístico

P es el modelo estadístico alterado

m_k es el factor de aleatoriedad

f es el factor de intensidad

y_k es el valor del modelo en un lapso de tiempo determinado

El factor de intensidad f determina la intensidad con la que la alteración del tráfico se aleja del tráfico presentado por el modelo estadístico. Una alteración muy pequeña daría como resultado un cambio poco significativo del tráfico obtenido con respecto al tráfico del modelo estadístico, por otro lado un factor f elevado ocasionaría un factor muy elevado de falsas alarmas en las pruebas de desempeño.

Durante las pruebas realizadas, se encontró que un factor f entre 0.1 y 0.25 propor-

Algoritmo 5 Alteración del modelo estadístico

ENTRADA: Modelo estadístico correspondiente al nodo raíz

SALIDA: Modelo estadístico alterado

1. Para cada dato y_k en el modelo
 - a) Calcular valor del factor r_k
 - 1) $ran = rand(2)$
 - 2) Si $ran = 1$, $r_k = -1$
 - 3) Otro, si $ran = 2$, $r_k = 1$
 - b) Calcular valor del factor f
 - 1) $ran = rand(3)$
 - 2) Si $ran = 1$, $f = 0.25$
 - 3) Otro, si $ran = 2$, $f = 0.2$
 - 4) Otro, si $ran = 3$, $f = 0.1$
 - c) Calcular alteración del valor del modelo
 - 1) $m_k = r_k \times f \times y_k$
 - 2) $y_k^p = y_k + m_k$
 - 3) $y_k = y_k^p$
-

ciona una alteración muy realista de los modelos estadísticos dinámicos, dando como resultado la obtención de un comportamiento de red artificial realista.

Este procedimiento es efectuado a nivel del nodo raíz de la topología de red. De tal manera que al alterar el modelo estadístico dinámico del dispositivo de interconexión raíz, automáticamente los demás modelos se ven afectados.

A continuación se presenta el procedimiento usado para la alteración del modelo estadístico dinámico, el cual es mostrado en el algoritmo 5.

4.6.2. Generación de tráfico de red artificial

Una vez que el modelo estadístico dinámico ha sido modificado, el siguiente paso consiste en producir el comportamiento de las direcciones IP, para lo cual es necesario generar la información correspondiente a las tablas *tot_par* y *gra24* de la base de datos del sistema monitor de red.

Los factores tomados en cuenta al momento de producir el comportamiento de las

Protocolo	Porcentaje
TCP	95 %
UDP	4 %
ICMP	<1 %
OTRO	<0.5 %

Cuadro 4.9: Porcentaje del uso de protocolos

direcciones IP como fueron: Número total de paquetes que entran y salen por el nodo raíz, número total de paquetes que entran y salen por cada dirección IP y protocolos utilizados por los paquetes circulantes.

El primer factor es dado por la alteración del modelo estadístico dinámico del nodo raíz, con el cual, es posible repartir el número de paquetes entrantes y salientes para cada dirección IP. El siguiente paso consiste en decidir que protocolos utilizará cada paquete utilizando el criterio mostrado en el Cuadro 4.9.

Los pasos usados para reproducir de manera artificial el uso de protocolos es mostrado en el algoritmo 6, en este se asigna un paquete aleatoriamente a cada dirección IP, este proceso continua hasta agotar los paquetes de red asignados por el modelo en el lapso Δt .

4.6.3. Generación de fallas y abusos artificiales

Para llevar a cabo la simulación de fallas en la red son necesarios los siguientes datos: Hora inicial y final, nombre del dispositivo de interconexión fallido en el caso de una falla y dirección IP en el caso de un abuso. El algoritmo 7 muestra la metodología usada para la simulación de fallas y abusos.

4.6.4. Generación de exploraciones de red artificiales

La simulación es efectuada tomando en cuenta que un paquete de red necesita los siguientes datos para poder comenzar su recorrido a través de la red: Direcciones IP fuente y destino y los números de puerto fuente y destino.

Basados en la información contenida en la tabla *tot_par*, es posible conocer la cantidad de paquetes correspondientes a cada una de las direcciones IP, con lo cual es posible generar todos los pares IP+puerto fuente y destino necesarios para cada paquete artificial simulado.

Algoritmo 6 Generación de tráfico de red artificial

ENTRADA: Modelo estadístico dinámico alterado, direcciones IP de la topología

SALIDA: Llenado de las tablas *tot_par* y *gra24* de la base de datos del sistema monitor de red

1. Para cada x_k
 - a) Para cada y_k
 - 1) Asignar una paquete de red a una dirección IP aleatoriamente
 - b) Para cada dirección IP
 - 1) Mientras $cont \leq$ número de paquetes
 - $ran = rand(100)$
 - Si $ran \leq 96$, asignar TCP
 - Si $97 \leq ran \leq 98$, asignar UDP
 - Si $ran = 99$, asignar ICMP
 - Si $ran = 100$, asignar OTRO
 1. Asignar información del modelo alterado en la tabla *gra24*
 2. Asignar información de paquetes de red y protocolos en la tabla *tot_par*
-

Algoritmo 7 Generación de fallas y abusos

ENTRADA: Tipo de comportamiento anómalo, hora inicial y final, nombre del dispositivo de interconexión fallido o dirección IP

SALIDA: Incorporación del trafico anómalo al generador de tráfico de red artificial

1. Si $hora\ inicial \leq h \leq hora\ final$
 - a) Si falla
 - 1) Encontrar todas las direcciones IP hijas del dispositivo seleccionado
 - 2) Para cada dirección IP
 - a' Eliminar tráfico
 - b) Otro si abuso
 - 1) Aumentar n veces el tráfico circulante de la dirección IP
 2. Incorporar tráfico anómalo al generador de tráfico de red artificial
-

Una manera sencilla de generar comportamientos de red es mantener fijos tres de los cuatro datos que conforman una cuádrupla de envío de los paquetes de red mostradas en la sección 4.2. Al efectuar esta operación durante un lapso de tiempo Δt , se logra obtener una simulación de exploración de red.

Este método es ilustrado en el algoritmo 8.

En conclusión el simulador desarrollado, permite la creación de escenarios realistas de circulación de paquetes dentro de una red de área local, ya que, este brinda todos los escenarios útiles para llevar a cabo todas las pruebas necesarias en la etapa de desarrollo del sistema de supervisión automática de redes de área local, Teniendo como resultado final la capacidad de generar los siguientes escenarios de comportamiento:

- Simulación de tráfico libre de comportamientos anómalos
- Simulación de tráfico presentando la falla de un dispositivo de interconexión en un lapso de tiempo determinado
- Simulación de tráfico presentando un abuso en el uso de los recursos de la red por parte de una dirección IP en un lapso de tiempo determinado
- Simulación de tráfico presentando un exploración de puertos en un lapso de tiempo determinado
- Simulación de tráfico presentando un exploración de direcciones IP en un lapso de tiempo determinado

Cabe destacar que en una sola sesión de simulación de tráfico pueden presentarse fallas o abusos y exploraciones de red, dependiendo el escenario deseado a generar.

En resumen, el sistema desarrollado ofrece una manera sencilla, eficaz y de bajo costo para realizar la detección de comportamientos anómalos en el tráfico de red. La detección está basada en el análisis del comportamiento individual de los dispositivos de interconexión de la red usando para esto los modelos estadísticos. Como complemento es realizado a su vez un análisis para la detección de intentos de intrusión, realizando búsquedas de exploraciones de puertos y direcciones IP. Diversas pruebas de desempeño y eficacia fueron desarrolladas gracias al tráfico artificial generado por el simulador de red.

Algoritmo 8 Generación de exploraciones de red

ENTRADA: Datos de la tabla *tot_par*SALIDA: Datos para la tabla *datos*

1. Mientras existan paquetes
 - a) Si es comportamiento normal
 - 1) Obtener aleatoriamente una dirección IP fuente
 - 2) Obtener aleatoriamente un número de puerto fuente
 - 3) Obtener aleatoriamente una dirección IP destino diferente
 - 4) Obtener aleatoriamente un número de puerto destino
 - 5) Insertar datos en la tabla *datos*
 - 6) Disminuir en 1 el número de paquetes de salida de la dirección IP fuente
 - 7) Disminuir en 1 el número de paquetes de entrada de la dirección IP destino
 - b) Otro, si es un exploración de puertos
 - 1) Obtener aleatoriamente números de puerto destino diferentes
 - 2) Insertar datos en la tabla *datos*
 - 3) Disminuir en 1 el número de paquetes de salida de la dirección IP fuente
 - 4) Disminuir en 1 el número de paquetes de entrada de la dirección IP destino
 - c) Otro, si es un exploración de direcciones IP
 - 1) Obtener aleatoriamente una dirección IP destino
 - 2) Insertar datos en la tabla *datos*
 - 3) Disminuir en 1 número de paquetes de salida de la dirección IP fuente
 - 4) Disminuir en 1 el número de paquetes de entrada de la dirección IP destino
-

Capítulo 5

Interfaz de usuario

Toda la información acerca del comportamiento de la red es presentada al administrador de la red mediante una interfaz Web, la cual, tiene como propósito principal mostrar de manera sencilla y secuencial toda la información disponible acerca del comportamiento de la red de área local. Dicha información es mostrada por medio gráficas y tablas interactivas.

Las gráficas permiten observar de manera rápida el comportamiento de los paquetes circulantes tanto generales como por dispositivos de interconexión. Por otro lado, las tablas tienen como función mostrar todos los datos disponibles acerca de los paquetes de red que circulan a través de cada dirección IP. En la Figura 5.1 puede observarse la estructura de la interfaz Web desarrollada.

La interfaz Web fue desarrollada utilizando el lenguaje PERL-CGI, el cual facilita el desarrollo de páginas Web dinámicas. Además, debido a que el sistema de supervisión de redes locales también fue desarrollado utilizando el lenguaje PERL, se contaba con una compatibilidad completa entre los dos sistemas. La interfaz Web se encuentra instalada utilizando el servidor APACHE.

Además de la interfaz Web se cuenta con un editor de topologías de red denominado TKETOP, con el cual se incorpora la información de la topología de red a la base de datos del sistema. TKETOP fue desarrollado en lenguaje PERL en conjunto con Tk para el desarrollo de la interfaz gráfica.

Mediante el uso de ambas interfaces se ofrece al usuario de una vista completa sobre el comportamiento y estructura de la red supervisada ofreciendo las siguientes facilidades:

Gráficas de comportamiento. Mediante el uso de gráficas es mostrado el compor-

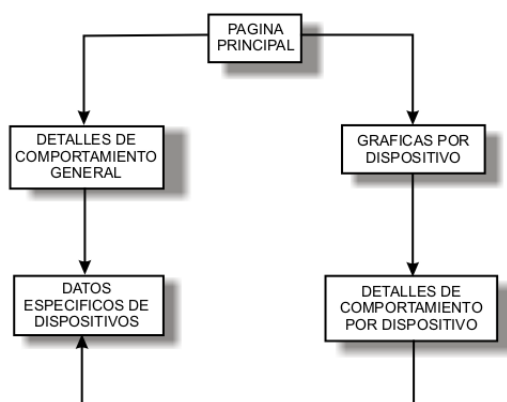


Figura 5.1: Estructura del sistema Web

tamiento real y esperado del tráfico de red, contando con gráficas de tráfico de entrada y salida y de dispositivos de interconexión.

Gráficas de error. Mediante estas gráficas es mostrada la diferencia entre el comportamiento real y el esperado.

Gráficas de protocolos. Tienen el propósito de mostrar el uso de los protocolos por cada dispositivo de interconexión.

Detalles de tráfico. Muestra el número de paquetes y protocolos usados por cada dirección IP.

Edición de topología. Tiene como propósito ofrecer una interfaz gráfica para la edición de la topología de la red a supervisar.

Mapa. Muestra de manera gráfica la topología de red ofreciendo detalles de cada dispositivo.

Reporte de comportamientos anómalos. Muestra los detalles del comportamiento de la red al detectarse un comportamiento anómalo.

En las secciones siguientes se explica con más detalle el funcionamiento de cada una de las partes que componen la interfaz de usuario.



Figura 5.2: Gráfica del comportamiento del tráfico de entrada mostrada en la interfaz Web

5.1. Interfaz Web

En la página principal de la interfaz Web, el administrador de red puede tener una vista general del comportamiento de la red. En esta página es mostrado en forma de gráficas el comportamiento de todo el tráfico que entra y sale por la red de área local. También es mostrada la diferencia resultante de la comparación entre el tráfico presentado y el esperado por el modelo estadístico.

5.1.1. Tráfico de entrada y salida

El tráfico de entrada y salida de la red es mostrado en la página principal mediante dos gráficas. Las dos gráficas comparten las mismas características. Una imagen de la gráfica del tráfico de entrada de la red, mostrada en la interfaz Web, es presentada en la Figura 5.2.

La gráfica ilustra los datos del comportamiento de la red mostrando la cantidad de paquetes de entrada de la red en un tiempo determinado. En este caso, se muestra el comportamiento total de paquetes de entrada que presentó la red durante un día entero. Esto quiere decir que tanto la gráfica de tráfico de entrada como la de salida, muestran el comportamiento general de la red, sin tomar en cuenta el comportamiento individual

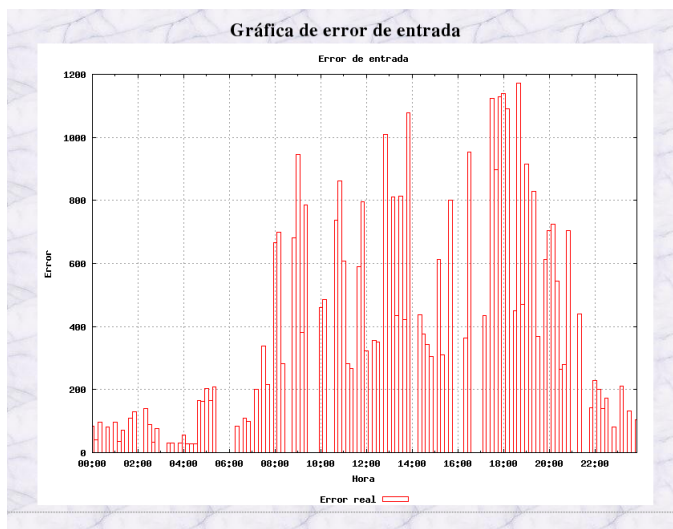


Figura 5.3: Gráfica de error entre el tráfico de entrada real y el tráfico de entrada esperado por el modelo estadístico

de los dispositivos de interconexión.

Estas gráficas cuentan con la característica de ser interactivas. Al presionar una determinada zona de la gráfica se muestra la página correspondiente a los detalles de comportamiento de las direcciones IP dentro del lapso de tiempo Δt seleccionado.

Error de tráfico de entrada y salida

Junto con las dos gráficas del comportamiento de entrada y salida, la página principal contiene dos gráficas adicionales. Estas corresponden a la diferencia obtenida de la comparación entre el tráfico de red real y el tráfico de red esperado por el modelo estadístico. La gráfica de error del tráfico de entrada es mostrada en la Figura 5.3, la cual muestra la cantidad de paquetes en los que difiere el tráfico real y el esperado en un lapso de tiempo determinado. La función que cumplen estas gráficas es la de mostrar de una manera rápida y sencilla el error que se presenta entre el tráfico real y el esperado por el modelo estadístico. De esta manera el administrador puede darse una idea del desempeño actual de la red. Cabe mencionar que este error corresponde a la comparación realizada entre todo el tráfico de red que entra o sale de la red, según sea el caso y el tráfico general de la red esperado por el modelo estadístico.

Reporte de comportamiento por hora

Día	Hora	Total	IP	Entrada	Salida	TCP	UDP	ICMP	Otro
0713	1300	425	141.20.2.6	208	217	355	44	12	14
0713	1300	446	141.20.2.10	243	203	386	34	12	14
0713	1300	419	141.20.1.3	237	182	356	39	9	15
0713	1300	448	141.20.2.11	240	208	367	52	11	18
0713	1300	476	141.20.2.9	270	206	388	53	17	18
0713	1300	418	141.20.2.4	212	206	333	47	12	26
0713	1300	429	141.20.2.7	238	191	356	44	14	15
0713	1300	465	141.20.1.1	238	227	388	44	19	14
0713	1300	425	141.20.3.1	222	203	362	39	13	11
0713	1300	460	141.20.3.3	240	220	380	50	15	15
0713	1300	472	141.20.2.3	265	207	379	53	20	20
0713	1300	469	141.20.3.2	268	201	387	50	17	15
0713	1300	467	141.20.1.2	252	215	375	54	12	26
0713	1300	499	141.20.2.8	274	225	419	48	13	19
0713	1300	436	141.20.2.5	254	182	379	36	13	8
0713	1310	480	141.20.2.6	267	213	387	61	15	17
0713	1310	509	141.20.1.3	291	218	431	49	16	13
0713	1310	462	141.20.2.10	275	187	378	50	20	14
0713	1310	478	141.20.2.11	261	217	398	48	17	16
0713	1310	474	141.20.2.9	273	201	410	40	16	8
0713	1310	445	141.20.2.4	248	197	371	44	11	19
0713	1310	498	141.20.3.1	312	186	421	47	14	16
0713	1310	508	141.20.1.1	278	230	427	50	11	20
0713	1310	471	141.20.2.7	266	205	400	42	14	15

Figura 5.4: Tabla de detalles de comportamiento general

5.1.2. Detalles de tráfico en la red local

Detalles de direcciones IP

En esta página son mostrados los detalles del comportamiento del tráfico de cada una de las direcciones IP presentes en el lapso de tiempo Δt seleccionado. Los detalles son mostrados en forma de tabla, la cual contiene los siguientes campos: día, hora, total de paquetes circulantes, dirección IP, paquetes de entrada y salida, Paquetes transmitidos por protocolos TCP, UDP, ICMP y otro.

Una fracción de la tabla de detalles del tráfico de entrada es mostrada en la Figura 5.4.

Tipo:	PC
Nombre:	PCX2
IP:	141.20.2.10
Número de puertos:	0
Conectado al puerto	1
Descripción	PC ubicada en la sala 3

Padre:	hub3
--------	----------------------

Hijos:

,

Ruta del dispositivo:

-> -> [141.20.2.1](#) -> [switch1](#) -> [141.20.2.2](#) -> [hub3](#) -> [141.20.2.10](#)

Figura 5.5: Datos específicos por dispositivo de red

Detalles de dispositivos de red

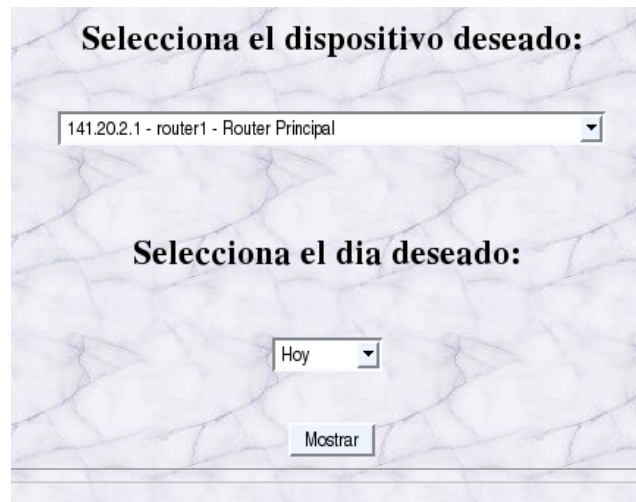
Al seleccionar el enlace de una dirección IP determinada en la tabla de comportamiento antes mencionada se muestran sus datos específicos. La Figura 5.5 muestra un ejemplo de dichos datos.

La página se encuentra dividida en cuatro partes. La primera de ellas está conformada por una tabla, la cual tiene como propósito proporcionar la información general del dispositivo de red seleccionado tales como: tipo de dispositivo de red, nombre del dispositivo, dirección IP, número de puertos y descripción.

En la segunda parte se muestra el dispositivo de interconexión padre del mismo. En este caso además de proporcionar el dato sobre el padre, este sirve como enlace a sus datos. Esto quiere decir que es posible ver los mismos datos específicos del padre en esta página.

En la tercera parte se encuentra la tabla correspondiente a los dispositivos de red hijos. De la misma manera que la parte anterior, cada nombre proporcionado funciona como enlace hacia su información en esta misma página.

Por último en la cuarta parte, se proporciona la ruta del dispositivo seleccionado. Cada nombre de dispositivo de interconexión de la ruta mostrada, sirve a su vez como enlace hacia sus datos específicos



The image shows a web interface with a light blue background. At the top, it says "Selecciona el dispositivo deseado:" in bold. Below this is a dropdown menu with the text "141.20.2.1 - router1 - Router Principal". Underneath, it says "Selecciona el día deseado:" in bold. Below this is another dropdown menu with the text "Hoy". At the bottom, there is a button labeled "Mostrar".

Figura 5.6: Selección de dispositivos de interconexión

5.1.3. Tráfico por dispositivo de interconexión

Esta sección, permite al administrador obtener información del comportamiento del tráfico correspondiente a los dispositivos de interconexión de la red. En la Figura 5.6 se muestra una parte de la página de selección de dispositivos de interconexión.

En esta se muestran dos listas desplegables. La primera lista contiene todos los dispositivos de interconexión que conforman la red de área local. La segunda lista permite seleccionar el día de la semana del cual se desea información. Esta información es mostrada en forma de gráficas. En este caso, se usan dos clases de páginas. La primera muestra el comportamiento del tráfico de red de un dispositivo de interconexión. La Figura 5.7 muestra la gráfica correspondiente a un dispositivo de interconexión.

En la gráfica se muestra la cantidad de paquetes circulantes en un lapso de tiempo determinado. La gráfica muestra tres tipos de comportamiento: Con una línea continua se muestra el tráfico circulante total por la red. Es decir, el tráfico total entrante y saliente que pasa por el dispositivo raíz. Con una otra línea continua pero de menor magnitud se muestra el tráfico circulante total que circula por el dispositivo de interconexión seleccionado. El tercer y último comportamiento es mostrado en forma de impulsos, estos corresponden al modelo estadístico dinámico. Es decir, el tráfico circulante que se espera circule por el dispositivo de interconexión seleccionado.

Los dos primeros comportamientos se obtienen en base a la información proporcionada por el sistema monitor de red. El tercer comportamiento es obtenido por el

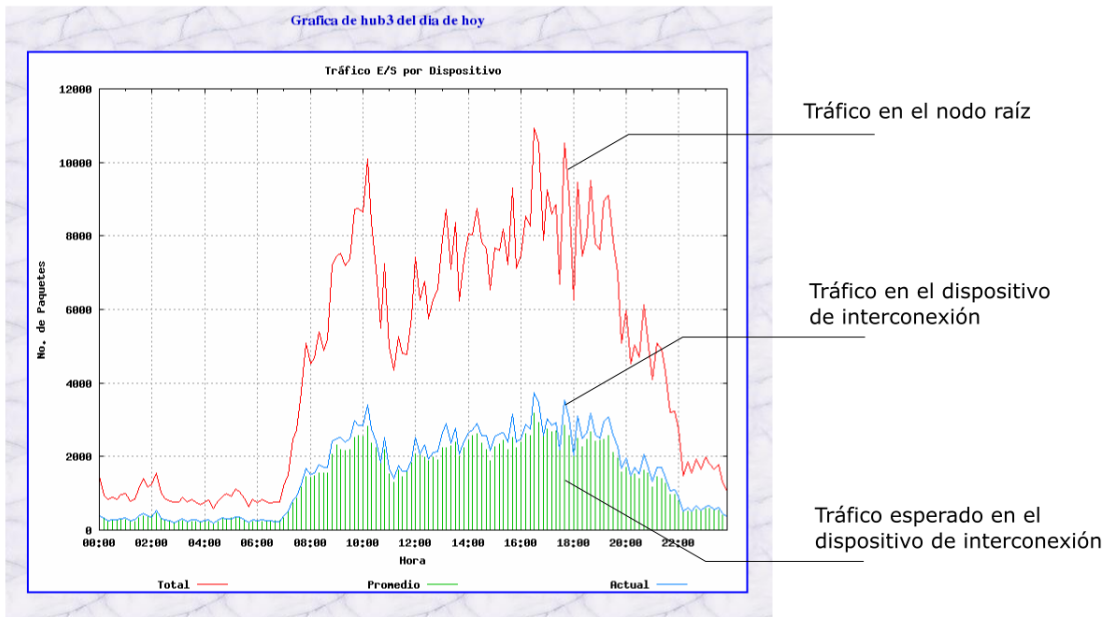


Figura 5.7: Gráfica de comportamiento de un dispositivo de interconexión

módulo de retroalimentación de modelos estadísticos dinámicos. Esta operación se efectúa utilizando los datos del segundo comportamiento.

Con esta gráfica el administrador de red es capaz de tener en una misma pantalla la posibilidad de comparar el tráfico circulante por el dispositivo de interconexión contra la cantidad total de paquetes que circula por toda la red de área local. Además de contar con la posibilidad de verificar el buen desempeño del dispositivo mediante la comparación de su comportamiento contra el comportamiento esperado por el modelo estadístico dinámico.

Esta gráfica comparte la misma característica interactiva de las gráficas de comportamiento del tráfico de entrada y salida de la Figura 5.2. Sin embargo, los detalles del comportamiento de las direcciones IP, corresponden a las direcciones IP hijas del dispositivo seleccionado. De esta manera el administrador sólo verá los detalles del comportamiento correspondiente solo al dispositivo de interconexión y no a toda la red.

Otra gráfica contenida en esta página corresponde a la mostrada en la Figura 5.8, la cual muestra el comportamiento del tráfico del dispositivo de interconexión. Pero a diferencia de la primera, en esta gráfica se muestra el número de paquetes que circulan por un protocolo determinado. Los protocolos mostrados son los siguientes: TCP, UDP, ICMP. Cualquier otro protocolo se muestra como OTRO dentro de la gráfica. Las dos

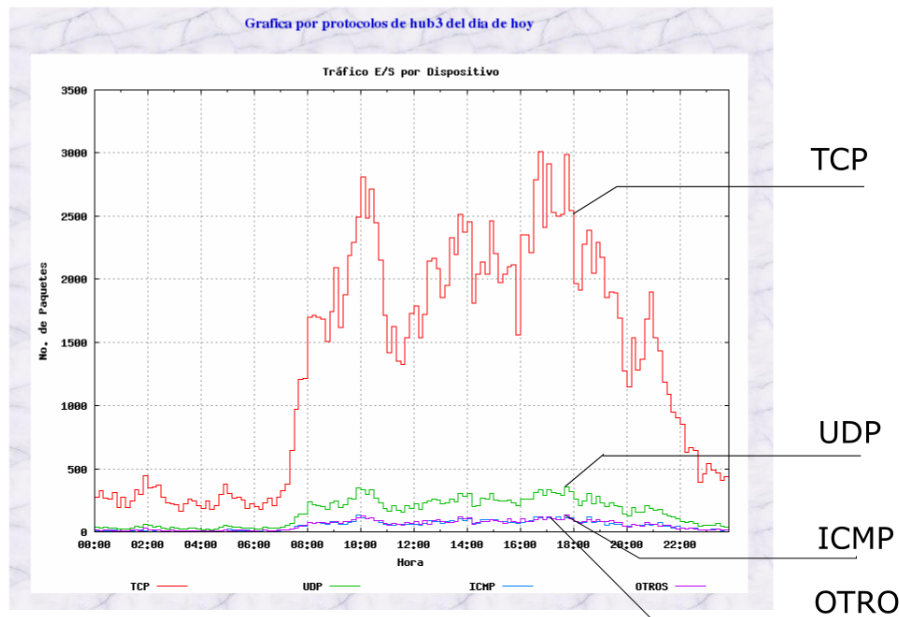


Figura 5.8: Gráfica de comportamiento de un dispositivo de interconexión por protocolos
clases de gráficas son actualizadas automáticamente cada 10 minutos.

5.1.4. Direcciones IP por dispositivo de interconexión

Al seleccionar una zona determinada en la gráfica de comportamiento por dispositivo de interconexión se muestra una página, la cual contiene los detalles del comportamiento del dispositivo seleccionado. Dicha tabla se muestra en la Figura 5.9.

Esta tabla comparte las mismas características de la tabla mostrada en la Figura 5.4. Sin embargo la diferencia radica en que en esta tabla no se muestran todas las direcciones IP de la red. En esta tabla sólo se muestra la información correspondiente a las direcciones IP hijas del dispositivo de interconexión seleccionado. Con esto se elimina mucha información que, tal vez no resulte de utilidad en ese instante al administrador de red.

Las direcciones IP mostradas en el campo IP de la tabla funcionan como enlaces. Estos enlaces dirigen al administrador de red a la página mostrada en la sección 5.1.2. En esta página se despliega toda la información correspondiente a la dirección IP seleccionada y a todos los dispositivos de red relacionados con la misma.

Comportamiento de tráfico de hub3

Día	Hora	Total	IP	Entrada	Salida	TCP	UDP	ICMP	Otro
0713	0900	505	141.20.2.10	329	176	417	55	16	18
0713	0900	473	141.20.2.11	292	181	418	35	13	7
0713	0900	452	141.20.2.9	278	174	380	42	16	15
0713	0900	544	141.20.2.4	352	192	448	55	22	19
0713	0900	520	141.20.2.5	321	199	445	46	13	16
0713	0910	487	141.20.2.10	287	200	408	51	16	13
0713	0910	478	141.20.2.11	269	209	403	46	17	12
0713	0910	555	141.20.2.9	295	260	473	48	21	13
0713	0910	523	141.20.2.4	305	218	415	66	22	21
0713	0910	485	141.20.2.5	276	209	408	49	14	14
0713	0920	460	141.20.2.10	196	264	386	46	12	16
0713	0920	481	141.20.2.11	197	284	415	38	14	14
0713	0920	456	141.20.2.9	207	249	372	48	18	18
0713	0920	490	141.20.2.4	219	271	418	45	12	15
0713	0920	506	141.20.2.5	234	272	420	53	16	17
0713	0930	496	141.20.2.10	270	226	422	46	12	17
0713	0930	498	141.20.2.11	284	214	406	53	20	19
0713	0930	512	141.20.2.9	305	207	415	56	22	19
0713	0930	486	141.20.2.4	274	212	392	57	17	20

Figura 5.9: Tabla de detalles de comportamiento por dispositivo de interconexión

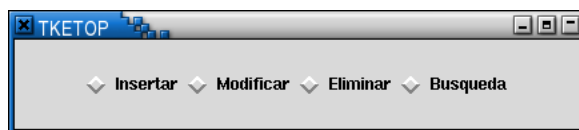


Figura 5.10: Menú principal de TkETOP

5.2. Editor de topología TkETOP

El editor desarrollado cumple el propósito de almacenar la información de la topología dentro de la base de datos del sistema supervisor de red. El editor fue desarrollado utilizando el lenguaje PERL y la interfaz gráfica Tk. TkETOP permite la ejecución de cuatro operaciones principales: Insertar, Modificar, Eliminar, y Búsqueda. La Figura 5.10 muestra el menú principal de TkETOP.

Inserción

La ventana insertar contiene tres partes como lo muestra la Figura 5.11, la primera ubicada en la parte superior de la ventana contiene todos los tipos de elementos que pueden ser contenidos en la red. En la parte izquierda de la ventana se muestra una lista que contiene todos los dispositivos de interconexión A la derecha de la ventana se

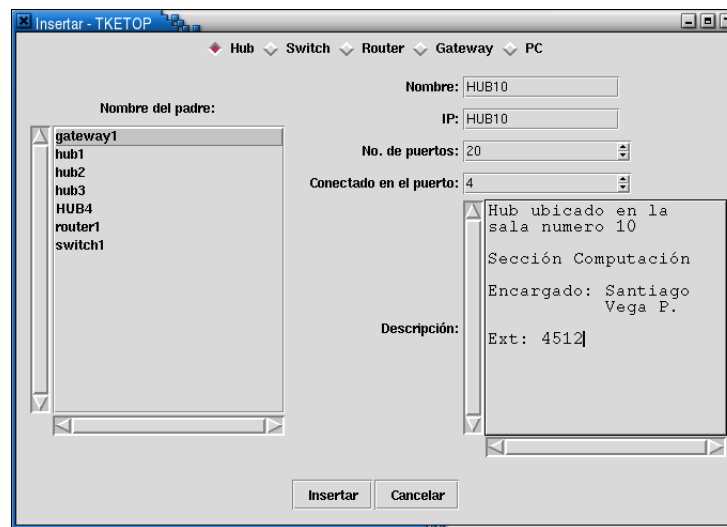


Figura 5.11: Ventana Insertar

ubicar los campos que corresponden a los datos del elemento a insertar.

Modificación

La ventana modificar mostrada en la Figura 5.12 esta formada por dos partes. La primera es una lista que contiene todos los elementos contenidos en la red. En la parte derecha se muestran todos los campos que pueden ser modificados, dependiendo del tipo de componente éstos podrán o no ser habilitados para su modificación.

Supresiones

La ventana eliminar esta constituida por una lista donde se muestran todos los dispositivos de la red. Los dispositivos de interconexión sólo pueden ser eliminados si no existen elementos que estén conectados a ellos. Ver Figura 5.13.

Búsquedas

La ventana de búsqueda está formada por cinco partes como lo muestra la Figura 5.14, la primera localizada en la parte superior nos muestra un conjunto de listas desplegables mostrando todos los dispositivos según su tipo. En la parte izquierda de la ventana se encuentran dos listas de elementos, en la primera lista se muestra el dispositivo padre, la segunda lista muestra todos los dispositivos hijos correspondientes. En la

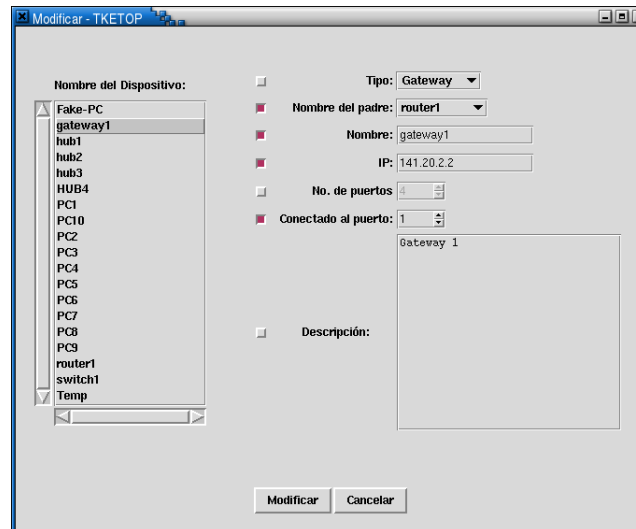


Figura 5.12: Ventana Modificar

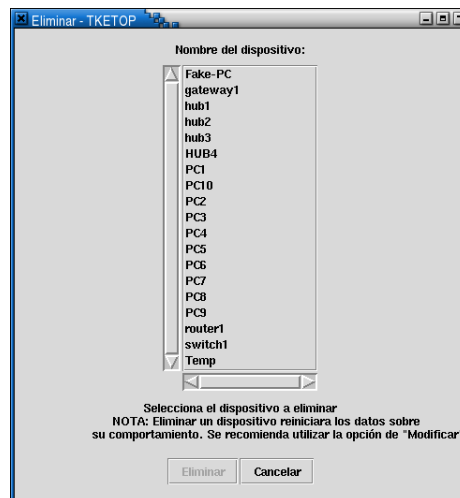


Figura 5.13: Ventana Eliminar

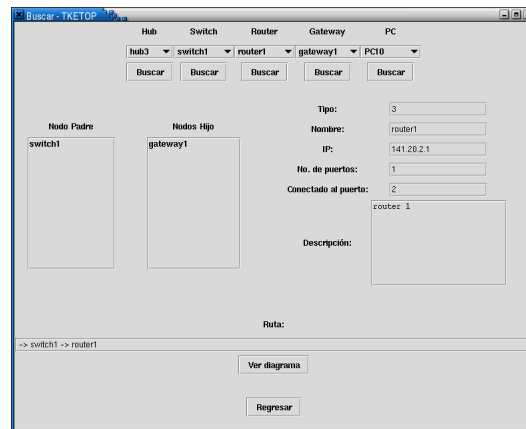


Figura 5.14: Ventana Búsqueda

parte derecha de la ventana se muestra toda la información del elemento seleccionado, los campos son similares a los contenidos en la ventana Insertar. La quinta parte de la ventana localizada en la parte inferior muestra la ruta de conectores, desde el conector raíz hasta el conector seleccionado.

Generación de mapas

El mapa de la Figura 5.15 nos muestra un ejemplo de los mapas generados, cada dispositivo es representado por un cuadro. Dentro de cada cuadro se encuentra contenida la información correspondiente a cada elemento: dirección IP, tipo, nombre, número de puertos, y a que puerto de conexión. Cada nodo está conectado a la red por medio de líneas que representan las conexiones físicas de la topología, al presionar con el ratón sobre una de ellas cambia su color, indicando su punto de partida y destino. Al cambiar los datos o la conexión de un elemento a otro conector, el mapa es automáticamente actualizado.

El trabajo más cercanamente relacionado es TkIned [17, 16]. TkIned muestra iconos para cada uno de los diferentes nodos, y puede desplegar gráficas de los valores obtenidos de dichos nodos. Sin embargo no soporta agregación de la información mostrada.

Otra herramienta relacionada es XNetMod [6]. XNetMod soporta la agregación de topologías, conectores, máquinas y subredes. Así como también realiza un análisis estadístico de la red. Sin embargo, esta herramienta solo está dirigida a la simulación de redes sin que tenga la capacidad de ser implementada en una red real.

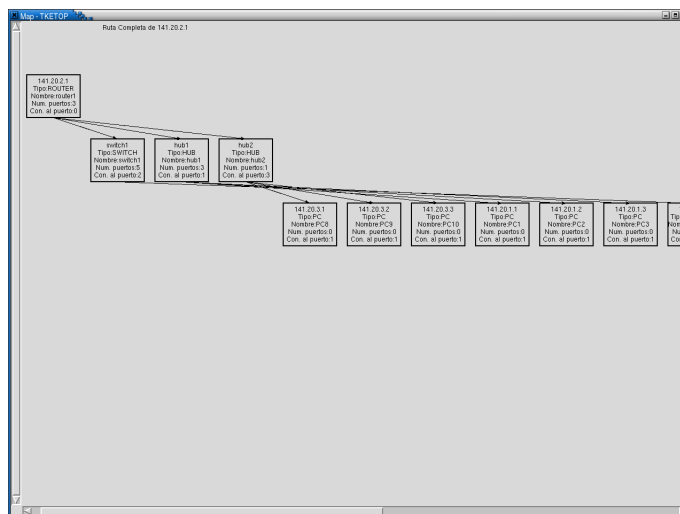


Figura 5.15: Mapa de la topología generado por TkETOP

5.3. Reportes de comportamientos anómalos

Además de las interfaces antes mostradas el sistema supervisor muestra de manera automática reportes de fallas, abusos e intentos de intrusión en la red a través de la terminal. Para ejemplificar dichos reportes utilizaremos la red mostrada en la Figura 5.16.

Reportes de fallas y abusos

Para ejemplificar el reporte generado por el sistema al presentarse una falla utilizaremos el comportamiento obtenido de la simulación de una falla en el dispositivo de interconexión “SWITCH1”. La gráfica de dicho comportamiento es mostrada en la Figura 5.17. La falla ocurre desde la 10 hasta las 12 horas. Debido a que el dispositivo fallido es un dispositivo padre los efectos de la falla afectaron a los dispositivos de interconexión: GATEWAY1, HUB3 Y HUB4.

En la Figura 5.18 se muestra el reporte generado por el sistema supervisor de red durante la simulación de dicha falla. El reporte incluye la información correspondiente a la hora de la falla, dispositivos de interconexión anómalos, dispositivos de interconexión fuente y el comportamiento del tráfico real y esperado de cada dispositivo afectado.

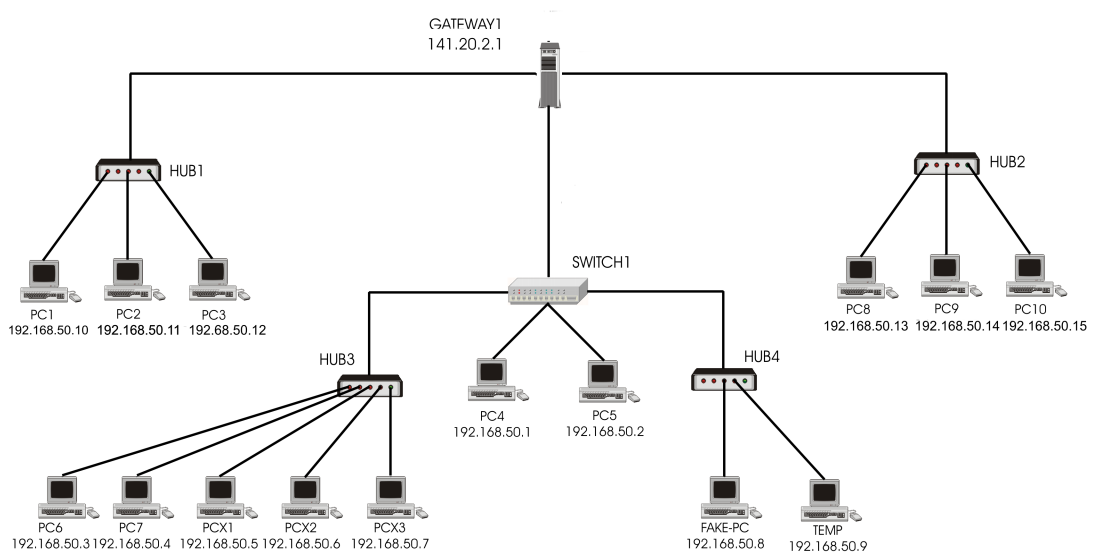


Figura 5.16: Red de área local simulada.

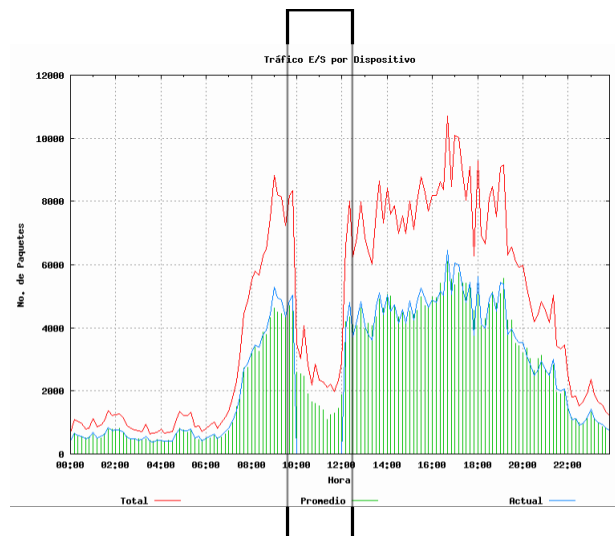


Figura 5.17: Gráfica de comportamiento de SWITCH1

```

File Edit View Terminal Tabs Help
1000
-----FAILS-----
Falla en switch1 a las 1000 con un comportamiento 0-5192
Falla en 141.20.2.2 a las 1000 con un comportamiento 0-5284
Falla en hub3 a las 1000 con un comportamiento 0-2543
Falla en HUB4 a las 1000 con un comportamiento 0-1395
} Dispositivos anómalos

Rastreando....
Posibles nodos anómalos:
-switch1
} Dispositivos anómalos fuente

1010
-----FAILS-----
Falla en 141.20.2.1 a las 1010 con un comportamiento 3023-9183
Falla en switch1 a las 1010 con un comportamiento 0-5109
Falla en 141.20.2.2 a las 1010 con un comportamiento 0-5194
Falla en hub3 a las 1010 con un comportamiento 0-2562
Falla en HUB4 a las 1010 con un comportamiento 0-1318

Rastreando....
Posibles nodos anómalos:
-switch1

1020
-----FAILS-----
Falla en switch1 a las 1020 con un comportamiento 0-4935
Falla en 141.20.2.2 a las 1020 con un comportamiento 0-5017
Falla en hub3 a las 1020 con un comportamiento 0-2451
Falla en HUB4 a las 1020 con un comportamiento 0-1262

Rastreando....
Posibles nodos anómalos:
-switch1

```

Figura 5.18: Reporte de fallas y abusos

```

File Edit View Terminal Tabs Help
Virtual_Ports
-----intruders.pl-----
SCAN_PORTS: ..... [OK]
SCAN_IPS: ..... [OK]
0700
Virtual_Ports
-----intruders.pl-----
----SCAN_PORTS----
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 25 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 80 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3219 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3871 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 1524 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3524 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 35250 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 35513 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3265 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 80 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3267 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3269 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3272 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3278 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 25 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 4292 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 36478 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3293 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3306 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 4194 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3721 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3021 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3892 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3723 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3333 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3389 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 42411 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 1884 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3156 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3625 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3971 HORA: 0700
Posible exploración de puertos con IP fuente: 192.168.50.10 , IP destino: 192.168.50.13 , desde el puerto fuente: 36478 con puerto destino: 3219 HORA: 0700
SCAN_IPS: ..... [OK]

```

Figura 5.19: Reporte de exploración de puertos

Reporte de intentos de intrusión

El reporte de intentos de intrusión incluye información sobre posible exploraciones de puerto y direcciones IP. En dichos reportes se muestra las direcciones IP fuente, las direcciones IP destino, los números de puerto origen y los números de puerto destino involucrados en la exploración. La Figura 5.19 muestra el reporte obtenido de la simulación de una exploración de puertos que tuvo lugar a las 7 horas.

En conclusión, la interfaz Web desarrollada permite al administrador de red tener tanto una vista general como específica del comportamiento del tráfico de la red. Debido a que, tanto se puede consultar la información correspondiente a toda la red en conjunto como el comportamiento de una dirección IP de un dispositivo de interconexión específico. La incorporación de los modelos estadísticos por dispositivo de interconexión a la interfaz Web, proporciona al administrador de una base de comparación. Debido a que gracias a los modelos, el administrador es capaz de tener un punto de comparación para la toma decisiones sobre el desempeño de la red.

TKETOP puede interactuar con el mantenimiento o aplicaciones de administración de la red. Trabajando en conjunto con sistemas monitores de red se puede extender la capacidad de análisis en la búsqueda de anomalías en la red. El haber desarrollado el sistema usando Perl/Tk proporciona facilidad para probar nuevas adiciones a nuestro

sistema, incluso con el sistema corriendo. Además de hacerlo portable, flexible y fácil de modificar.

Capítulo 6

Conclusiones y trabajo futuro

La supervisión automática de redes de área local es un problema que se encuentra aún en estudio debido a su complejidad. Existen diversas herramientas que logran detectar comportamientos anómalos presentados en una red de este tipo. Aunque ya existen muchas herramientas que logran detectar la mayoría de las anomalías presentes en la red, estas varían en gran manera en exactitud y costo.

El presente trabajo contribuye al estudio de la detección automática de comportamientos anómalos en una red de área local mediante la propuesta de un método de bajo costo para la detección y localización de anomalías. Esta reside en el análisis del comportamiento del tráfico presentado por los distintos dispositivos de interconexión que conforman la red de área local. Ya que un análisis a nivel de dispositivos de interconexión resulta más exacto que uno efectuado solo a nivel de puerta de enlace. Para efectuar dicho análisis es necesario contar con la información topológica de la red.

Se construyó una herramienta capaz de introducir la información topológica sobre la red de área local mediante una interfaz gráfica. En conjunto con esta herramienta y la información obtenida por medio de un sistema monitor de red, es posible realizar un análisis detallado sobre el comportamiento del tráfico de la red supervisada.

Los modelos estadísticos permiten tener una predicción del tráfico que circulará por los dispositivos de interconexión en un lapso de tiempo determinado, y con base a la predicción dada, realizar una detección de anomalías en la red de área local. Los modelos estadísticos son dinámicos, esto quiere decir que los modelos son capaces de diferenciar entre un comportamiento anómalo y un cambio normal presentado en el comportamiento del tráfico de red. De esta manera se disminuye el rango de falsas alarmas. Al presentarse un cambio normal en el comportamiento de la red, los modelos

estadísticos dinámicos son actualizados automáticamente, lo cual les añade la ventaja de ser auto-suficientes. Por lo cual, el administrador de la red no se ve en la necesidad de construir un conjunto de modelos nuevos cuando un cambio en la red es efectuado.

Modelos estadísticos dinámicos correspondientes a cada uno de los dispositivos de interconexión, proporcionan la información necesaria para la detección de comportamientos anómalos en distintas zonas de la red de área local. Debido a que se cuenta con un modelo por cada dispositivo de interconexión es posible para el sistema determinar automáticamente cual dispositivo de interconexión es el culpable del comportamiento irregular presentado por la red. De esta manera se obtiene un nivel más detallado y exacto de detección de comportamientos anómalos. Aunque dichos modelos no son extremadamente precisos debido a su naturaleza estadística, son lo suficientemente adecuados para cubrir el objetivo de la detección de las fallas y abusos más graves a los cuales una red de este tipo puede sufrir durante su periodo de vida. La característica presentada por los modelos estadísticos dinámicos al no ser muy exactos, le proporcionan a la etapa de análisis la ventaja de poder detectar en periodos cortos fallas y abusos presentados por los dispositivos de interconexión.

Para compensar la falta de exactitud de los modelos estadísticos dinámicos se cuenta con un método capaz de detectar intentos de intrusión a la red, tales como exploraciones de puertos y exploraciones de direcciones IP. Al complementar los modelos estadísticos dinámicos con este módulo, se obtiene un mayor nivel de detección de comportamientos anómalos. Los comportamientos anómalos que puede detectar la herramienta son: *fallas, abusos, ataques, gusanos, exploraciones de puerto y de direcciones IP*.

Una interfaz Web permite realizar consultas sobre el estado de la red de área local de manera sencilla. La interfaz cuenta con la capacidad de mostrar el comportamiento del tráfico actual de cada dispositivo de interconexión y de las computadoras correspondientes a cada uno de ellos. Además se cuenta con la capacidad de consultar la información del comportamiento que fue presentado por los dispositivos de interconexión durante una semana de actividad.

Es posible acoplar al sistema con un método de descubrir e introducir de manera automática la topología de la red de área local. Ya que la herramienta desarrollada requiere realizar la introducción de la topología de manera manual. El sistema desarrollado cuenta con la capacidad de detectar y avisar al administrador de red sobre comportamientos anómalos presentados en la red. Por lo tanto, se puede agregar un método capaz de tomar decisiones y efectuar acciones de manera automática con el objetivo de corregir o detener de manera automática la fuente del mal funcionamiento

de la red de área local. El tipo de comportamientos anómalos detectados por el sistema ayuda al administrador de red a tener un mayor control de la seguridad de la red. Sin embargo, existen comportamientos anómalos los cuales no pueden ser detectados, entre ellos se encuentran aquellos que no pueden ser detectados mediante métodos estadísticos. Un ejemplo de estos incluyen generadores de claves, enmascaramientos, penetraciones de usuarios legítimos, caballos de Troya y virus [8].

Bibliografía

- [1] Uyles Black. *TCP/IP and related protocols*. Uyles Black series on computer communications, 1992.
- [2] Martin C. Brown. *PERL Manual de referencia*, chapter 7, pages 192–199. Osborne McGraw-Hill, 2001.
- [3] Bill Cheswick, Hal Burch, and Steve Branigan. Mapping and visualizing the internet. pages 1–12, citeseer.ist.psu.edu/cheswick00mapping.html, 2000.
- [4] K. C. Claffy. Measuring the internet. *IEEE Internet Computing*, pages 73–75, Jan-Feb 2000.
- [5] M. Becker Debar Herve and D. Siboni. A neural network component for an intrusion detection system. In *IEEE Symposium on Reserach in Security an Privacy*, Oakland CA, May 1992.
- [6] K. Deboo. Xnetmod: A design tool for large-scale networks. 96-6, CITI, August 1993.
- [7] David L. Stevens Douglas E. Comer. *Internetworking with TCP/IP*. Prentice-Hall, 1993.
- [8] Denning Dorothy E. An intrusion detection model. *IEEE Symposium on Security and Privacy*, Abril 1986.
- [9] Lemmonier E. Protocol anomaly detection in network-based ids. In *Defcom, Sweden, Stockholm*, June 2001.
- [10] A. Díaz E. Morfín. Análisis del tráfico de una red local. Master’s thesis, CINVESTAV-IPN, Septiembre 2004.

- [11] Lunt Teresa F. A survey of intrusion detection techniques. *Computers an Security*, 12:4, June 1993.
- [12] Paul J. Fortier. *Handbook of LAN Technology*, chapter 7, pages 145–183. McGraw-Hill, 2nd edition, 1992.
- [13] J. Reed Fox K. L., R Henning. A neural network approach toward intrusion detection. In *13th National Computer Security Conference*, Washintong DC, October 1990.
- [14] Josep Fábrega Francesc Comellas. *Metamática discreta*, chapter 5, pages 123–130. Universitat Politècnica de Catalunya, Barcelona, España, 2002.
- [15] R. A. Kemmerer G. Vigna. Netstat: A network-based intrusion detection approach. In *Proceedings of the 14th Annual Computer Science Applications Conference*, Scottsdale, AZ, December 1998.
- [16] H. Langendörfer J. Schönwälder. How to keep track of your network configuration. In *LISA VII Conference*, 1993.
- [17] H. Langendörfer J. Schönwälder. Ined an application independent network editor. In *SANS II*, Abril 1993.
- [18] D. E. Johnson. Applied multivariate methods for data analysis. *Brooks/Cole Publishing Co.*, 1998. Duxbury.
- [19] Sandeep Kumar. *Classification and Detection of Computer Intrusions*. PhD thesis, Purdue, IN, 1995.
- [20] K. N. Levitt B. Mukherjee J. Wood L. T. Heberlein, G. V. Dias and D. Wolber. A network security monitor. In *In Proceedings of the IEEE Symposium on Research in Security and Privacy*, Oakland, CA, April 1990.
- [21] J.C. Laprie. *Dependability: Basic cocepts and terminology*. Springer-Verlag, 1991.
- [22] Averill M. Law and W. David Kelton. *Simulation Modeling and Analysis*. McGraw-Hill, 2000.
- [23] Miljenko Mikuc Marko Zec. Real-time ip network simulation at gigabit data rates. In *7th Intl. Conference on Telecommunications (ConTEL)*, June 2003.

- [24] Mark A Miller. *LAN Protocol Handbook*. M and T Books, 1990.
- [25] M. Stolarchuk M. Sienkiewicz A. Lambeth E. Wall M.J. Ranum, K. Landfield. Implementing a generalized tool for network monitoring. In *Proceedings of 11th Syst. Admin. Conf. (LISA 97)*, pages 1–20, 1997.
- [26] Tomas Olovsson. A structured approach to computer security. Technical report, Chalmes University of Technology, 1992.
- [27] P. A. Poras P. G. Neumann. Experiences with emerald to data. In *Proceedings of 11th Usenix Workshop on Intrusion Detection and Network Monitoring*, pages 11–12, Santa Clara, CA, April 1999.
- [28] Kihong Park. On the effect and control of self-similar network traffic: A simulation perspective. In *Winter Simulation Conference*, pages 989–996, cite-seer.ist.psu.edu/article/park97effect.html, 1997.
- [29] David Plonka Paul Barford, Jeffery Kline and Amos Ron. A signal analysis of network traffic anomalies. In *Proceedings of ACM SIGCOMM Iinternet Measurement Workshop*, 2002.
- [30] V. Paxson. Experiences learned from bro. *The Usenix Assoc. Magazine*, pages 21–22, September 1999.
- [31] Charles P. Pfleeger. *Security in computing*. Prentice Hall, 1997.
- [32] Katherine E Price. Host-based misuse detection and conventional operating systems audit data collection. Master’s thesis, Purdue University, December 1997.
- [33] W. Mendenhall III R. L. Scheaffer and R. L. Ott. Elementary survey sampling. *Wadsworth Publ. Co.*, 1996.
- [34] Red-IRIS. Incidentes de seguridad año 2004. <http://www.rediris.es/cert/doc/informes/2004/node4.html>, Enero 2005.
- [35] Patrick Regan. *Local Area Networks*, chapter 2, pages 112–115. Prentice Hall, 1st edition, March 2003.
- [36] M. Roesch. Snort - lightweight intrusion detection for networks. In *In 13th Systems Administration Conference - LISA 99*, 1999.

- [37] Garfinkel S. and E.H. Spafford. *Practical Unix and Internet Security*. O'Reilly and Associates, second edition edition, 1996.
- [38] D. M. Teal S. E. Smaha, T. Grance and D. Mensur. Dids - motivation, architecture, and an early prototype. In *Proceedings of 14th National Computer Security Conference*, pages 167–176, Washington, DC, October 1991.
- [39] B. Ralph S. Northcutt, V. Irwin. Shadow. *Naval Surface Warfare Center Dahlgren La*, 1998.
- [40] R. Crawford M. Dilger J. Frank J. Hoagland K. Levitt C. Wee R. Yip D. Zerkle S. Stanfield-Chen, S. Cheung. Grids - a graph-based intrusion detection system for large networks. *The 19th National Information Systems Security Conference*, pages 361–370, October 1998.
- [41] C. Bor-Sen P. Sen-Chueh and W. Ku-Chen. Traffic modeling, prediction, and congestion control for high-speed networks: A fuzzy approach. *IEEE Transaction on Fuzzy Systems*, 8(5):491–508, 2000.
- [42] S. E. Smaha. Haystack: An intrusion detection system. In *Proceedings IEEE Fourth Aerospace Computer Science Applications Conference*, Orlando, Fl, December 1998.
- [43] C. Lares V. Jacobson and S. McCanne. tcpdump. *LBNL, University of California*, June 1997.
- [44] W. Willinger W. E. Leland, M. S. Taqqu and D.V. Wilson. On the self-similar nature of ethernet traffic (extended version). *IEEE/ACM Transaction on Networking*, 2(1):1–45, 1994.
- [45] C. Williamson. Internet traffic measurement. *IEEE Internet Computing*, pages 70–74, Nov-Dic 2001.