CARACTERIZACIÓN DE CONDICIONES DE FALLA SOBRE UN SEGMENTO DE RED, BASADA EN LAS VARIABLES DE LA MIB

GIOVANNI BRACHO TOVAR
LUIS DANIEL SARMIENTO GONZÁLEZ

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES
BUCARAMANGA
2005

CARACTERIZACIÓN DE CONDICIONES DE FALLA SOBRE UN SEGMENTO DE RED, BASADA EN LAS VARIABLES DE LA MIB

GIOVANNI BRACHO TOVAR
LUIS DANIEL SARMIENTO GONZÁLEZ

Este proyecto es presentado como requisito para optar al título de Ingeniero Electrónico

Director
OSCAR GUALDRÓN GONZÁLEZ, Ph.D.
Codirector
PAOLA GUZMÁN, M.Sc.

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES
BUCARAMANGA
2005

AGRADECIMIENTOS

Los autores expresan su agradecimiento y reconocimiento a:

Nuestras familias por su apoyo incondicional.

La Especialización en Telecomunicaciones.

La Escuela de Ingeniería Eléctrica, Electrónica y Telecomunicaciones

La Universidad Industrial de Santander.

TABLA DE CONTENIDO

[1] Local Area Network: Red de Área Local

LISTA DE TABLAS

Pág.

Tabla 13. Resultados de detección en los tres algoritmos implementados para el grupo de variables de la falla de Tormenta de Broadcast.

LISTA DE FIGURAS

Pág.

LISTA DE ANEXOS

Pág.

TITULO

CARACTERIZACIÓN DE CONDICIONES DE FALLA SOBRE UN SEGMENTO DE RED, BASADA EN LAS VARIABLES DE LA MIB[*]

AUTORES

Giovanni Bracho Tovar
Luis Daniel Sarmiento González[**]

Palabras Claves

MIB, Gestión, Falla, Metodología.

Descripción

Una falla puede causar interrupciones de servicio y daños irremediables a una red. Así el desarrollo de un sistema efectivo de gestión de fallas es una tarea fundamental. Una de las herramientas más importantes para la creación de tal sistema, es la información contenida en las variables que conforman las bases de información de gestión de los dispositivos de red (MIB).

Para aprovechar esta herramienta, es necesario seleccionar las variables que aportarán la información para suplir las necesidades de gestión. En la literatura, se presentan frecuentemente procedimientos de selección de variables, que generalmente siguen criterios basados en la experiencia de los investigadores. Este planteamiento, es difícil de seguir, ya que no existe un consenso sobre cuales son los criterios a seguir para la selección y clasificación de estas variables para el mejor aprovechamiento de la información que almacenan.

Aunque se han realizado muchos trabajos en esta área, no existe aún, una metodología clara y general que establezca estos criterios y se pueda seguir para afrontar el diseño e implementación de herramientas que cumplan con la tarea de realizar la gestión de fallas en una red. En este aspecto radica la importancia de este trabajo pues su objetivo es la creación de una base para la realización de una metodología general, que permita llegar a la selección de variables MIB, críticas para la gestión, mediante la asociación de la información que almacenan, con las características de desempeño que identifican la posible ocurrencia de fallas en el segmento de red.

---

[*] Trabajo de Grado

[**]Facultad de Ingenierías Físico-Mecánicas, Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones, Director de Proyecto: Oscar Gualdrón González Ph.D

TITLE

CHARACTERIZATION OF FAULT CONDITIONS ON A NETWORK SEGMENT BASED IN THE MIB VARIABLES

AUTHORS

Giovanni Bracho Tovar
Luis Daniel Sarmiento González

Keywords

MIB, Management, Fault, Methodology.

Abstract

A fault can cause interruptions on services and irremediable damages to a network. Thus the development of an effective system of fault management becomes a fundamental task. One of the most important tools for the creation of such system is the information contained in the variables that does part of the management information bases of the network devices (MIB).

In order to make good use of this tool, it is necessary to perform a selection of variables that will provide the necessary information to supply the management requirements. The literature often present procedures of variable selection that generally follow researcher's experience based criteria. This proposal it is difficult to follow, because do not exist until now a general agreement about which are the criteria to follow to carry out a selection and classification of these variables for the best use of information that they store.

Although many works in this area have come true, do not exist still, a general and simple methodology that establish these criteria and may be followed to confront the design and implementation of tools that perform the management of network faults. Within this aspect resides the importance of this work because its objective is the creation of a basis for the accomplishment of a general methodology, that allow reaching the selection of MIB'S variables critical for the management by means of the association of information that they store, with the performance characteristics that identify the possible occurrence of faults in the network area.

INTRODUCCIÓN

El crecimiento acelerado de las redes de datos -y en especial de las LAN, ha originado un gran interés en la gestión de los recursos de red. La tendencia actual es hacia redes de mayor tamaño y por lo tanto de mayor complejidad, debido a esto, y a la gran dependencia que de ellas tienen las organizaciones en la actualidad las actividades de gestión se han convertido en un factor crítico. La calidad de servicio se ha convertido en un aspecto clave en cualquier red; por lo que el personal encargado de la gestión debe estar en capacidad de detectar, diagnosticar y corregir los problemas rápidamente, en lo posible antes de que impacten en la comunidad de usuarios.

La mayoría de estas tareas hasta ahora, han sido altamente dependientes de la actividad humana; pero el gran volumen de datos y la demanda de rapidez en la realización de estas actividades han hecho evidente la necesidad de buscar soluciones automatizadas. Dentro de las actividades de administración de redes, la gestión de fallas tiene una alta prioridad, ya que una falla simple puede propagarse a través de la red y causar pérdidas considerables.

Un baluarte a la hora de realizar las tareas de gestión y que puede convertirse en factor clave en el momento del diseño de aplicaciones automatizadas, es el conjunto de variables MIB[4], presentes en los dispositivos administrables dentro de una red (routers, switches, servidores, etc.), la MIB es una base de datos que acumula información relacionada con la configuración, estado y estadísticas del dispositivo, aspectos que pueden ser asociados con el comportamiento de la red.

El estudio propuesto establece los criterios de selección de variables MIB con base en las fallas de redes Ethernet típicas, como parte del planteamiento de una metodología general que permita establecer las posibles condiciones de falla propias de un segmento de red.

[4] Management Information Base: Base de información de gestión.

1. FUNDAMENTOS SOBRE GESTIÓN DE REDES Y BASES DE INFORMACIÓN DE GESTIÓN

En este capítulo se presentan los conceptos básicos sobre las características y elementos que constituyen las redes de datos de área local (LAN), basadas en la tecnología Ethernet; además se realiza una breve introducción a las nociones de gestión de redes, sus diversas funciones, y los protocolos y herramientas que hacen parte del modelo gestor-agente.

También son presentadas las definiciones generales de las fallas que afectan comúnmente a las redes basadas en tecnología Ethernet.

1.1 CONCEPTOS GENERALES

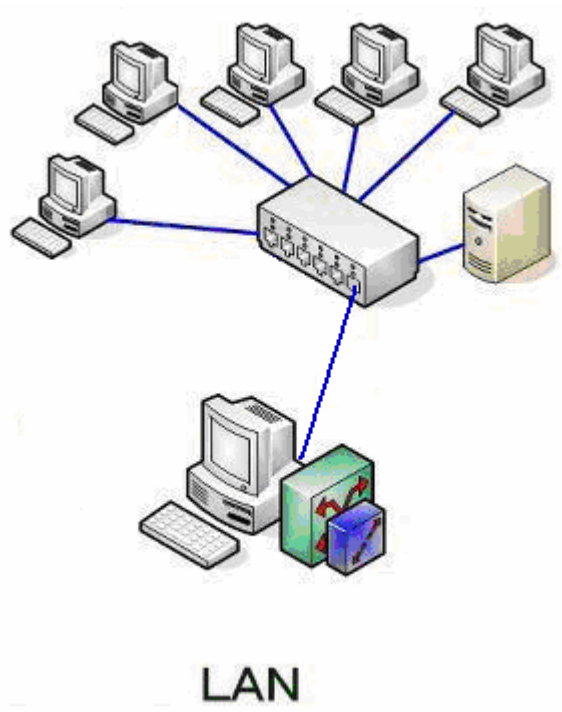
1.1.1 Red de datos de área local (LAN[5]). Es una red de datos de alta velocidad que cubre un área geográfica relativamente pequeña. Conecta típicamente estaciones de trabajo, computadores personales, impresoras, servidores, y otros dispositivos.

Las LAN ofrecen a los usuarios de computadores muchas ventajas, incluyendo el acceso compartido a los dispositivos y a las aplicaciones, intercambio de archivos entre los usuarios conectados, y la comunicación entre los usuarios vía correo electrónico, entre otras.

Los computadores que no son servidores de archivos reciben el nombre de estaciones de trabajo. Estos suelen ser menos potentes y tienen software personalizado para cada usuario. La mayoría de las redes LAN están conectadas por medio de cables y tarjetas de red, una en cada equipo.

[5] Local Area Network: Red de Área Local

Figura 1. Esquema general de una LAN



1.1.2 Tecnología Ethernet. A mediados de los años 70, la compañía Xerox desarrolló un método para transmitir datos entre computadores sobre un único cable. Esta tecnología se denominó ***Ethernet***. Mediante esta tecnología, podían conectarse muchos computadores a la misma red para crear una conexión de alta velocidad para la transferencia de datos.

Ethernet es un estándar de capa física y de enlace de datos definido por el grupo de protocolos **IEEE[6] 802.3**, que están basados en las tecnologías desarrolladas por las compañías Xerox, DEC, e Intel y que se ha convertido en la referencia para las redes de área local (LAN).

Estos protocolos definen el tamaño de datagrama, las reglas de conexión, los tipos y longitudes de cableado, las velocidades de transmisión, etc.

[6] Institute of Electric and Electronic Engineers: Instituto de Ingenieros Eléctricos y Electrónicos.

Un segmento físico, se refiere a un dominio de colisión, o a la extensión de cableado donde rigen las normas de longitud y temporización de colisiones de Ethernet. Si las estaciones de un segmento quieren acceder a la red escuchan si hay alguna transmisión en curso y si no es así transmiten.

Este estándar utiliza un mecanismo de acceso múltiple y de detección de colisiones, conocido como **(CSMA/CD)** [7] para regular el acceso al medio, ya que en el caso de que dos estaciones detecten la posibilidad de emitir y lo hagan al mismo tiempo, se producirá una colisión; pero esto queda resuelto con los sensores de colisión que detectan este problema y abortan o detienen la transmisión.

Las implementaciones del estándar Ethernet incluyen tres categorías principales:

- **Ethernet.** Es una especificación bandabase[8] para redes de area local que opera a 10 Mbps, usando CSMA/CD para funcionar sobre cable coaxial, par trenzado y fibra óptica. Fue creada en 1970, pero el término es usado frecuentemente para referirse a todas las LAN CSMA/CD. Esta categoría fue diseñada para operar en redes con requerimientos esporádicos de tráfico pesado.

- **Fast-Ethernet.** Es una tecnología de redes de area local de alta velocidad que opera a 100-Mbps sobre cableado de par trenzado o fibra óptica.

- **Gigabit-Ethernet.** Es una extensión del estándar 802.3 e incrementa la velocidad a 1000 Mbps o 1 Gbps esta promete ser una tecnología dominante para las LAN de alta velocidad y funciona sobre fibra óptica y cables de par trenzado.

[7] Carrier Sense Multiple Access/Collision Detection.

[8] Sistema de comunicación donde la señal es traspasada directamente de manera digital, especialmente para transmisiones cortas.

1.2 DISPOSITIVOS DE RED

La constitución de una red requiere la presencia de equipos especializados que permitan la interconexión entre las estaciones de trabajo o computadores y a su vez con las redes externas. Los dispositivos que generalmente cumplen estas funciones son: servidores, Routers (enrutadores), y Switches (conmutadores).

1.2.1 Servidores. Son computadores dedicados a servir de interlocutor entre todos los usuarios. Tienen altas prestaciones tanto en la velocidad de procesamiento de datos como en la capacidad de almacenamiento de estos. En los servidores se centralizan los recursos y aplicaciones de la red, reciben las peticiones de las terminales, las atienden y devuelven el resultado; Además son los encargados de ejecutar el software especial, llamado sistema operativo de red, empleado para administrar los recursos de ésta. Por su operación se clasifican en:

- **Servidores de comunicaciones.** Realizan todas las operaciones de comunicaciones requeridas por los usuarios.

- **Servidores de archivos.** Controlan los accesos a los archivos ubicados en un disco duro compartido.

- **Servidores de impresión.** Administran las colas de impresión.

- **Servidores de base de datos.** Manejan la administración de una base de datos común.

- **Servidores de correo.** Distribuyen el correo electrónico.

Un servidor central puede efectuar todas estas operaciones o bien delegar la realización de una determinada tarea al servidor específico correspondiente.

Los términos CLIENTE y SERVIDOR se utilizan tanto para referirse a programas que cumplen esas funciones, como para los computadores que las ejecutan, es decir, al computador que solicita un servicio se le llama cliente y al que ofrece dicho servicio se le denomina servidor.

1.2.2 Routers (Enrutadores). Un enrutador es un dispositivo de *propósito general* cuya función principal es la interconexión de redes, además tiene la propiedad de segmentar la red, con la idea de limitar el tráfico de difusión (broadcast) y proporcionar seguridad, control y redundancia entre dominios individuales de broadcast.

Estos equipos pueden filtrar protocolos y direcciones a la vez. Los equipos de la red saben que existe un router y le envían los paquetes directamente a él cuando se trata de equipos en otro segmento de red. Además los routers pueden interconectar redes distintas entre sí; eligen el mejor camino para enviar la información, balancean tráfico entre líneas, y brindan un acceso económico a una WAN[9].

El router trabaja con tablas de enrutamiento. A partir de la información que es generada por los respectivos protocolos de enrutamiento, toman decisiones como: ¿hay que enviar un paquete o no?, ¿cuál es la mejor ruta para enviar un paquete?, ¿cual es la mejor ruta para enviar la información de un equipo a otro?, etc.

1.2.3 Switch (Conmutador). Un switch registra la dirección física o de hardware de los computadores que se han conectado a él. Cuando recibe un mensaje, sólo lo envía al destinatario específico. Los switches cortan el tráfico de transmisiones innecesarias.

Estos equipos se utilizan así mismo para interconectar segmentos de red, (amplía una red que ha llegado a su máximo, ya sea por distancia o por el número de equipos) y se utilizan

[9] Wide Área Network: Red de Área Extensa.

cuando el tráfico no es excesivamente alto en las redes pero interesa aislar las colisiones que se produzcan en los segmentos interconectados entre sí.

Los switches trabajan con direcciones físicas (MAC[10]), por lo que filtran tráfico de un segmento a otro. Esto lo hacen "escuchando" los paquetes que pasan por la red y así van configurando una tabla de direcciones físicas de equipos que tienen conectados a un lado y otro del segmento (generalmente tienen una tabla dinámica), de tal forma que cuando escucha en un segmento un paquete de información que va dirigido a ese mismo segmento no lo pasa al otro, y viceversa.

La última mejora en la tecnología de interconexión de redes, son los dispositivos híbridos que combinan la gestión de paquetes que realizan los routers y la velocidad de los switches, estos switches multicapa operan en las capas 2 y 3 del modelo de red OSI[11]. Las prestaciones de esta clase de dispositivos se diseñan para el núcleo de grandes redes institucionales. A veces son denominados switches/enrutadores o conmutadores IP[12]; buscan flujos de tráfico comunes, y conmutan estos flujos en la capa de hardware para lograr velocidad. Para el tráfico fuera del flujo habitual, los switches multicapa conmutan usando funciones de enrutadores. Esto permite que la alta carga de las funciones de enrutamiento sólo sean empleadas donde es realmente necesario, y emplea la mejor estrategia para la manipulación de cada paquete de la red.

1.3 INTRODUCCIÓN A LA GESTIÓN DE REDES

Un buen sistema de gestión de red debe conocer la diversidad de dispositivos existentes en ella y proporcionar un entorno apropiado para su administración. Así, los principios fundamentales de gestión dicen que:

[10] Medium Access Control: Control de Acceso al Medio.

[11] Open System Interconnection: Sistema de interconexión abierta.

[12] Internet Protocol: Protocolo del Internet.

"El impacto de realizar gestión de red a un nodo gestionable debe ser mínimo." [13]

"Cuando todo falla, la administración de red debe seguir funcionando".

La ISO[14] ha contribuido enormemente a la estandarización de las redes. Su modelo de gestión de redes es primordial para el entendimiento de las funciones principales de los sistemas de gestión de redes. Este modelo define cinco áreas conceptuales que encierran las distintas tareas de gestión.

1.3.1 Áreas funcionales de gestión. Esta clasificación es usada comúnmente al plantearse propuestas para el desarrollo de servicios de gestión.

- **Gestión de la configuración.** El objetivo de esta es monitorizar la información de configuración del sistema y los procedimientos relacionados con actualizaciones de software y hardware.

- **Gestión del rendimiento.** Su objetivo es el mantenimiento del nivel de servicio que ofrece la red a los usuarios, para asegurar una operación eficiente en todo momento. Se basa en actividades que evalúan de modo continuo los principales indicadores de desempeño de la red, y permite identificar cuellos de botella existentes y potenciales; y establecer tendencias adecuadas para la toma de decisiones y planeación del crecimiento.

- **Gestión de la seguridad.** Busca controlar el acceso a los recursos de la red, de acuerdo a las políticas locales de seguridad, para reducir la posibilidad de ataques y evitar que información crítica sea accesada por personas sin la autorización apropiada.

- **Gestión de la contabilidad.** Tiene como fin la medición y recolección de estadísticas

[13] CARRASCO, J; CAMPOS, J; RUIZ, C. "Gestión De Red: Protocolo SNMP" Grupo de Investigación en Señales, Telemática y Comunicaciones, GSTC. Universidad de Granada España, http://ceres.ugr.es/ 2004.

[14] Internacional Organization for Standardization: Organización Internacional para la Estandarización.

de utilización de los recursos de la red para lograr la adecuada distribución y asignación de las cuotas de uso de los usuarios individuales o grupales.

- **Gestión de fallas.**[15] Esta área funcional de administración de redes del modelo de gestión de la ISO tiene como objetivo fundamental la localización de los problemas de la red y su recuperación ante ellos. Es ideal la detección proactiva, pues de esta forma se identifican los cambios antes de que afecten el servicio a nivel de usuario y así asegurar un alto nivel de disponibilidad de los elementos y servicios del sistema de red.

Una definición de falla para redes basadas en tecnología Ethernet, que es comúnmente aceptada es: "Una falla es el conjunto de condiciones que hacen que el servicio entregado se desvíe del servicio prometido o pactado"; tomando como base este concepto es posible realizar una clasificación de fallas típicas que afectan a una red ethernet[16]

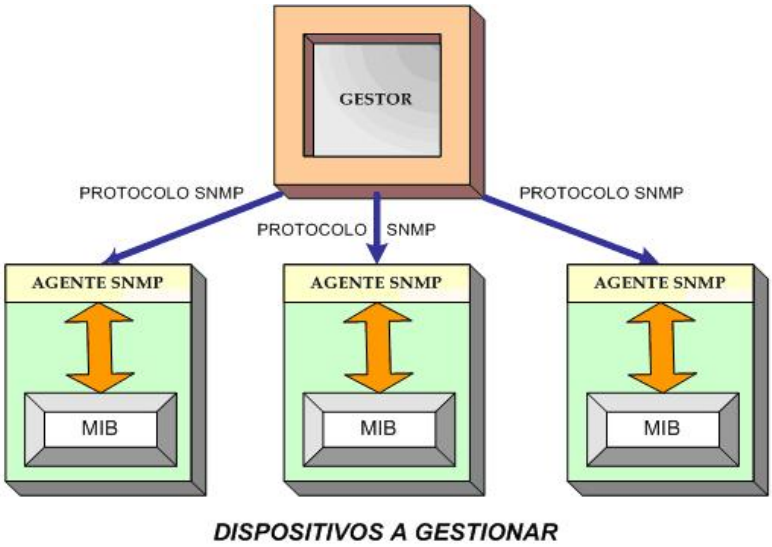
1.3.2 Modelo de gestión de redes. La arquitectura típica de los procedimientos de gestión de redes se basa en el modelo **Gestor-Agente**, que puede operar de dos formas. La forma directa consiste en un proceso de gestión centralizado en el cual una entidad llamada gestor obtiene la información de gestión interrogando a cada uno de los dispositivos administrables.

El otro esquema es la gestión distribuida en la cual los agentes presentes en cada dispositivo pueden comportarse también como gestores y recopilar información de otros dispositivos presentes en el mismo nodo de red y hacer un único reporte al gestor principal, ahorrando de esta manera recursos y tiempo en el proceso de gestión.

[15] THOTTAN, M; JI, Chuanyi. "Anomaly Detection in IP Networks". Bell Labs 2002.

[16] DELLA MAGIORA Paul; ELLIOT, Christopher. "Performance and Fault Management", Cisco press 2000.

Figura 2. Esquema general del modelo de gestión.



La arquitectura del modelo Gestor-Agente comprende los siguientes elementos.

- **Dispositivo gestionado.** Se denomina así a cualquier dispositivo de red (Router, Switch, Servidor de acceso, Hub, estaciones de trabajo o impresora) que contiene un agente y que reside en una red administrada. Esta clase de dispositivos obtiene y almacena información de gestión del nodo de red en el que están ubicados y la pone a disposición de los sistemas de monitorización de red (**NMS**[17]).

- **Gestor (Manager).** Es una entidad que posee un software llamado NMA[18] (aplicación de gestión de red), que posee una interfaz de operador para permitir a un usuario autorizado gestionar a los agentes existentes en los dispositivos gestionables que hacen parte de la red. El gestor hace uso de los protocolos de gestión para sondear la información, que se encuentra almacenada en los agentes que existen dentro de los dispositivos gestionables.

[17] Network Monitoring System: Sistema de Monitorización de Red.

[18] Network Monitoring Application: Aplicación de Gestión de Red.

- **Agentes.** Los agentes son módulos de software de gestión de redes que operan como interfaz con los dispositivos gestionables que hacen parte de la red. Estos incluyen sistemas finales que soportan aplicaciones de usuario o aquellos que ofrecen un servicio de comunicación, como controladores de switches, servidores y enrutadores. Un agente posee un conocimiento local de la información de gestión y la traduce a un formato compatible con el protocolo de gestión que se encuentre activo.

- **Objeto gestionado.** Los recursos de la red que pueden ser gestionados, son representados por este tipo de objetos o variables. Pueden ser definidos en términos de sus atributos, las operaciones a que puede ser sometido, las notificaciones que puede emitir y sus relaciones con otros objetos gestionados.

Existen dos clases de objetos gestionados, los escalares que representan un recurso de red simple y los tabulares que definen múltiples recursos de la red que están estrechamente relacionados, son agrupados en tablas formadas por objetos escalares y se encuentran condensados en las bases de información de gestión.

Un conjunto de objetos gestionados, junto con sus atributos, operaciones y notificaciones constituyen una base de información de gestión.

1.4 PROTOCOLOS DE GESTIÓN DE REDES

Uno de los primeros protocolos de gestión de redes, que ha sido usado exitosamente desde 1970 es el protocolo ICMP[19]. ICMP ofrece mecanismos para la transferencia de mensajes de control a partir de los enrutadores o las estaciones de trabajo, permitiendo una realimentación sobre los problemas en el ambiente de la red.

Este protocolo está disponible en todos los dispositivos, ya que es parte del esquema de

[19] Internet Control Message Protocol: Protocolo de Control de Mensajes del Internet.

protocolos TCP/IP[20]. La característica más útil del protocolo ICMP es el envío de mensajes de eco (petición/respuesta), que posibilitan la realización de pruebas para establecer si la comunicación entre dos entidades de red es posible. La mayoría de estos ecos son implementados a través de la utilidad PING[21] que está presente en la mayoría de sistemas operativos, y que fue una solución satisfactoria para la gestión de redes hasta el fin de los años ochenta.

El punto de partida para ofrecer un protocolo de gestión de redes específico fue el protocolo simple de monitorización de Pasarela (SGMP[22]), el cual suministró herramientas directas para monitorizar Gateways; además contemplaba un conjunto de principios e ideas que serían usadas luego en otros protocolos de gestión.

A medida que las necesidades de esquemas más sofisticados para la gestión de redes crecían, aparecieron varias propuestas que planteaban diferentes aproximaciones; una de ellas fue el protocolo simple de gestión de redes SNMP[23], el cual era una versión mejorada del SGMP.

Inicialmente el SNMP se adoptó como solución a corto plazo y el protocolo de información común de gestión (CMIP[24]), especificado por el modelo OSI y luego implementado como protocolo y servicio de información común de gestión sobre TCP/IP (CMOT[25]) fue considerado como la propuesta a largo plazo. Se estableció que tanto SNMP como CMOT tendrían la misma base de información para los objetos gestionados, pero pronto se hizo evidente que atar estos protocolos al nivel de objeto era totalmente

[20] Transmission Control Protocol/ Internet Protocol.

[21] Packet Internet Groper: Instrucción utilizada por el protocolo ICMP para verificar la conexión de hardware y la dirección lógica de la capa de red, el paquete típico de ping tiene un tamaño de 32 bytes de datos..

[22] Simple Gateway Management Protocol: Protocolo simple de monitorización de Pasarela.

[23] Simple Network Management Protocol: protocolo simple para la administración de red.

[24] Common Management Information Protocol: Protocolo de información de administración común.

[25] Common Management Information Service & Protocol over TCP/IP.

impráctico e inconveniente.

En el protocolo CMOT los objetos gestionados son vistos como entidades sofisticadas con atributos, procedimientos asociados y capacidades de notificación. Para mantener a SNMP lo más simple posible, se determinó no asociarlo con la base de gestión de CMOT; a partir de este momento la evolución del protocolo SNMP y su popularización iniciaron.

1.4.1 Protocolo SNMP. El protocolo Simple de Gestión de Red **SNMPv1** fue diseñado a mediados de los años 80 por Case, McCloghrie, Rose, y Waldbusser, como una solución a los problemas de comunicación entre diferentes tipos de redes.

En un principio, su principal meta era lograr una solución temporal hasta la llegada de protocolos de gestión de red más completos y con mejores diseños. Pero esos protocolos no llegaron y **SNMPv1** se convirtió en la única opción para la gestión de red. El manejo de este protocolo era simple, se basaba en el intercambio de información de red a través de mensajes conformados por unidades de datos de protocolo (**PDU**[26]).

Por ser un protocolo fácilmente extensible a toda la red, su uso se estandarizó entre usuarios y entidades; no obstante, este protocolo no era perfecto, además, no estaba pensado para poder gestionar la inmensa cantidad de redes que cada día iban apareciendo. Para subsanar sus carencias surgió la segunda versión (**SNMPv2**).

SNMP es un protocolo de capa de aplicación que facilita el intercambio de la información de gestión entre los dispositivos de la red. Es parte del grupo de protocolos que conforman la arquitectura (TCP/IP) y permite a los administradores de la red manejar su funcionamiento, detectar y solucionar problemas, así como planificar el crecimiento de la misma.

[26] Protocol Data Unit: Unidad de Datos de Protocolo.

1.5 BASES DE INFORMACIÓN DE GESTIÓN

La estructura de la información de gestión (***SMI***[27]) establece las reglas para definir la información de gestión y esta definida claramente en el protocolo SNMPv2. Si se piensa almacenar una colección de objetos gestionados, por ejemplo, en una base de datos, la SMI define el esquema de esa base de datos. Dichas bases de datos se conocen como bases de información de gestión o **MIB**.

Estas bases de datos administran información sobre el estado, configuración y desempeño de los dispositivos de red administrables.

En una red típica dispositivos como routers, switches, servidores y estaciones de trabajo mantienen una MIB que describe el estado de los recursos del sistema. La MIB, representa cada uno de los recursos del sistema mediante un objeto.

La MIB esta conformada por la información referente a cada objeto, y se define mediante una estructura de árbol, que le permite organizar dichos objetos en diversos grupos. Las ramas del árbol representan a los objetos gestionados, los cuales ofrecen información sobre algunos recursos de los dispositivos.

Estos objetos son consultados mediante operaciones de monitorización o sondeo y modificados a través de operaciones de control. En la estructura de árbol se define a los objetos con características afines dentro de grupos de objetos (object group).

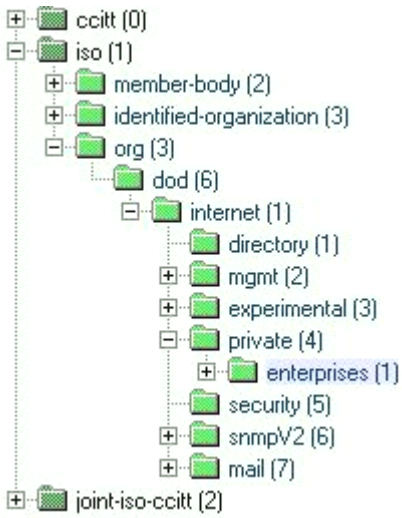
1.5.1 Identificadores de objeto. Un identificador de objeto (**OID**) es un nombre asignado arbitrariamente. Los OID pueden describirse como el nombre de los nodos del árbol, representan una secuencia de números enteros que están separados por puntos decimales, fijando de esta manera niveles que facilitan el seguimiento de la ruta hasta cualquier objeto dentro del árbol a partir de la raíz.

[27] Structure of Management Information: Estructura de Información de Gestión.

Para poder gestionar un objeto es necesario conocer su nombre, pero los objetos en sí no son más que plantillas, y son las instancias (índices, filas de una tabla) de los mismos, las que maneja el protocolo, el cual utiliza un identificador de objeto, formado por la concatenación del nombre del tipo de objeto y un sufijo, para identificarlas dentro de una tabla.

Los identificadores de objetos definidos en el SMI para el protocolo de gestión a partir de la raíz son:

Figura 3. Diagrama del árbol principal de la MIB.



Camino a la raíz

Internet **OID**= {ISO 3 6 1}

Directory **OID** = {Internet 1}

Mgmt **OID** = {Internet 2}

Experimental **OID** = {Internet 3}

Private **OID** = {Internet 4}

Enterprises **OID** = {private 1}

Security **OID** = {Internet 5}

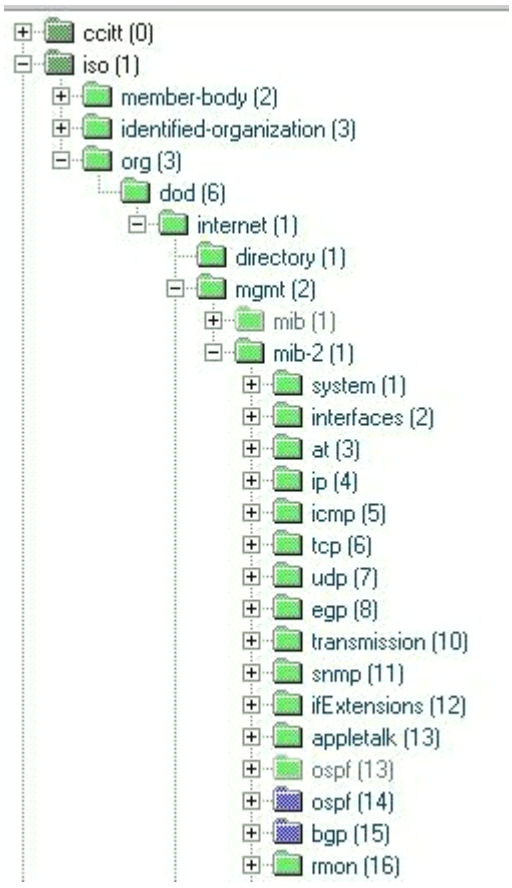
SNMPv2 **OID** = {Internet 6}

Mail **OID** = {Internet 7}

Sin embargo, el SMI no prohíbe la definición de objetos en otras porciones del árbol de objetos.

Las OID definidas para la gestión de redes siguiendo el modelo Internet fueron especificadas dentro de una rama específica del árbol dedicada a la administración de estas. El árbol extendido de la MIB con sus grupos estándar es presentado en la figura 4.

Figura 4. Diagrama Extendido del árbol de la MIB



1.5.2 Clases de módulos MIB. Cada conjunto de objetos gestionados se conoce como Módulo de Base de Información de gestión (Módulo MIB), existen tres tipos de módulos MIB:

- **Estándar**. Diseñados por un grupo de trabajo del **IETF** [28] y estandarizado por el **IESG**[29]. Los prefijos de los identificadores de objetos se encuentran bajo el subárbol ***mgmt***.

- **Experimental**. Mientras un grupo de trabajo desarrolla una MIB, los identificadores de

[28] Internet Engineering task force: Fuerza de Tareas de Investigación de Internet. Fuerza de tareas compuesta por más de 80 grupos de trabajo responsables por el desarrollo de estándares de Internet.

[29] Internet Engineering Steering Group: Grupo directivo de ingenieros en Internet. Organización nombrada por el IAB, que administra la operación de la IETF

objetos temporales se colocan bajo el subárbol ***experimental***. Si el MIB adquiere la condición de estándar, se colocan los identificadores bajo el subárbol ***mgmt***.

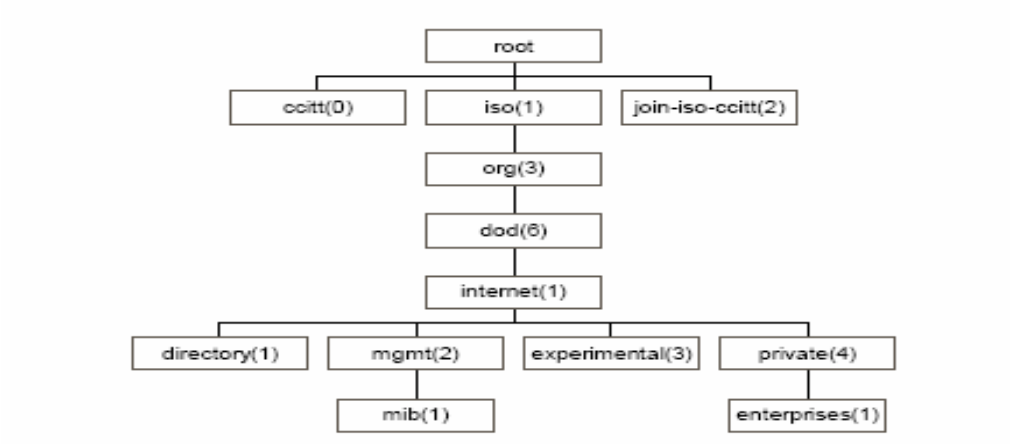
- **Específico**. La mayor parte de las empresas desarrollan módulos MIB propios que soportan ciertas características particulares, las cuales no son generalmente contempladas en los módulos MIB estándar.

1.5.3 Estructura de la MIB. En la estructura de árbol, el nodo relacionado con la información de SNMP es el nodo **Internet,** por lo que para establecer la organización de las variables de la MIB se deben seguir las ramificaciones de este nodo.

Este nodo se divide en los siguientes subárboles: **mgmt (2)** el cual usado para identificar objetos **"estándar"**, el subárbol **experimental (3)** es usado para identificar objetos diseñados por grupos de trabajo del IETF.

Si un módulo de información producido por un grupo de trabajo se convierte en un módulo de información "estándar", entonces en el momento de su entrada en los cauces estándar de Internet, los objetos se mueven al subárbol **mgmt (2)**. El subárbol **private (4)** se usa para identificar objetos definidos de forma unilateral. El subárbol **enterprises (1)** bajo el private se usa, entre otras cosas, para permitir a los proveedores de subsistemas de red registrar modelos de sus productos. El subárbol **SNMPv2** se usa con propósitos de mantenimiento.

Figura 5. Nodos principales del árbol de la MIB.



La MIB en su configuración estándar mantiene 171 variables distribuidas en los siguientes grupos[30]:

- **System (1).** Contiene objetos que describen alguna información básica sobre el agente SNMP, o el dispositivo de red en el que el software agente está siendo ejecutado.

- **Interface (2).** Ofrece información de configuración y estadísticas de desempeño de las interfaces de red. Esta información es aplicable a todo tipo de interfaz.

- **At (address translation) (3).** Este grupo era obligatorio para todos los sistemas pero fue descontinuado por la MIB-II, cada grupo de protocolo de red contiene sus propias tablas de traslación de direcciones.

- **IP (4).** Mantiene información importante relacionada con la operación del protocolo IP en el nodo de red.

- **ICMP (5).** Provee estadísticas sobre Mensajes ICMP, y es útil para administración de desempeño. Básicamente tiene contadores sobre diferentes tipos y condiciones de mensajes ICMP**.**

- **TCP (6).** Provee algoritmos, parámetros y estadísticas sobre el protocolo TCP. Supervisa segmentos enviados y recibidos, cantidad actual y acumulada de conexiones abiertas, estadísticas de errores, etc.

- **UDP (7).** Provee estadísticas de tráfico, y detalles sobre datagramas UDP y puntos extremos.

- **EGP (8).** Provee estadísticas de tráfico, y detalles sobre mensajes EGP generados,

[30] Ver figura 4. Árbol extendido de la MIB.

recibidos y no enviados, y condiciones de vecinos EGP.

- **Dot3 (transmisión) (9).** Contiene objetos relacionados con el medio de transmisión especifico para cada interfaz del sistema. Reservado para MIBs específicas de un medio físico.

- **SNMP (10).** Provee estadísticas de tráfico y operaciones del protocolo SNMP.

- **RMON (16).** Este grupo de objetos no hace parte de los grupos estándar iniciales. La supervisión remota (RMON[31]) es una especificación de monitorización estándar que permite a varios monitores de la red y sistemas de gestión, intercambiar datos de monitorización de red.

Las MIBs están definidas por diferentes documentos del IAB[32]. La definición de las recomendaciones y estándares de gestión se plantea en los RFC[33]; la versión actual del protocolo SNMP se definió mediante el RFC 1157.

La primera MIB fue definida en el RFC 1156, esta base de información era muy limitada ya que solo consistía de nueve grupos de variables. Esta base de información de gestión fue redefinida mediante el RFC 1213 y es conocida como MIB–II para Administración de Redes basadas en TCP/IP. La diferencia principal con la primera versión radica en que la MIB II refina objetos ya definidos en la MIB I, y agrega una gran cantidad de objetos nuevos, que representan un aumento en la flexibilidad de las funciones de gestión enfocándose primariamente a la gestión remota.

[31] Remote Monitoring: Especificación del agente MIB descrita en el RFC 1271 que define las funciones de monitorización remota de dispositivos de la red.

[32] Internet Architecture Board: Comité de Arquitectura de Internet. Comité de investigadores de internetworking que discute temas relativos a la arquitectura de Internet.

[33] Request For Comments:Petición de comentarios,Serie de documentos empleada como medio de comunicación primario para transmitir información acerca de la Internet. Algunas RFC son designadas por el IAB como estándares de Internet.

1.6 FALLAS DE RED

Generalmente los problemas en una red Ethernet se afrontan siguiendo la premisa de que las fallas de red se manifiestan a través de degradaciones del desempeño. El método más usado para notar estas degradaciones es la detección de anomalías, donde una anomalía es definida como "un desempeño estadísticamente inusual". A partir de esta definición se podrían orientar los esfuerzos para la descripción de fallas, hacia el análisis de las condiciones anómalas que las caracterizan.

1.6.1 Definición de falla. En la definición de una falla de red pueden presentarse varios criterios y opiniones, sobre todo en las LAN Ethernet. Una definición común establece que una falla de red consiste en el conjunto de condiciones que originan que el servicio entregado se desvíe del servicio especificado o esperado.

La gran mayoría de autores definen dos grandes clases de fallas en una red Ethernet: fallas severas y fallas leves.

- **Fallas severas.** Se caracterizan por la imposibilidad de entregar paquetes. Las causas posibles para una falla de este tipo son: falla de energía, un cable cortado o la falla de equipo de red principal (Ej. un enrutador). Este tipo de fallas son altamente perjudiciales, ya que el funcionamiento de la red se reduce a cero. La detección de estas fallas es sencilla, ya que son notadas rápidamente por los usuarios o los administradores de red.

- **Fallas leves.** Se caracterizan por una pérdida parcial de ancho de banda. Las fallas leves no están bien definidas, pero la literatura generalmente las caracteriza por la degradación del desempeño o pérdida de ancho de banda en la red. Las principales causas de las fallas leves son: uso inapropiado de la red, la congestión temporal que causa retraso de transmisión, las fallas de hardware en el servidor, las fallas de protocolos de nivel superior, o usuarios dañinos.

Las fallas leves también se pueden originar en la corrupción, pérdida o retraso excesivo de los datos. La pérdida ocasional de datos, conocida como errores transitorios, puede ser causada por interferencia electrostática, fallas de hardware intermitentes, o desbordamiento de la memoria en un servidor o un Switch.

1.6.2 Tipos de fallas. Los tipos de falla más comunes en redes Ethernet son:

- **Tormenta Broadcast (Broadcast storm).** Esta falla es causada por el envío repetitivo de mensajes de difusión (broadcast), usualmente en búsqueda de información o servicios. Además de perturbar el desempeño de todas las estaciones (ya que deben capturar el paquete broadcast), el desempeño de red se perturbará debido a la inundación de respuestas a la difusión (broadcast).

- **Congestión de red.** Esta falla ocurre cuando un incremento en la carga de red resulta en una disminución en la capacidad de trabajo útil de la misma. Bajo esta condición, una red de datos ha sido establecida bajo carga, en un estado en el que la demanda de tráfico es alta pero con un Throughput[34] muy pequeño, con altos niveles de pérdida de paquetes y retardos[35].

La congestión tiene varias causas, puede originarse en la retransmisión innecesaria de paquetes, o en los paquetes no entregados, que son transportados a través de la red pero se descartan antes de llegar a su destino final, produciendo un gran desperdicio de ancho de banda.

Las aplicaciones se convierten en una de las principales culpables de este tipo de falla. Un mal comportamiento de los paquetes, peticiones erróneas que bloquean los recursos del sistema, aplicaciones que no manejan de forma adecuada las peticiones de los clientes o

[34] Rendimiento total (de procesamiento), capacidad de ejecución, productividad.

[35] HIGBIE Carrie. "Congestion - Can Standards Provide Relief?" The Siemon Company 2004.

temporizadores mal establecidos, pueden originar este tipo de condiciones de falla.

También es común que se presente congestión cuando grandes incrementos en la carga ofrecida son correspondidos solamente con pequeños incrementos en el throughput de la red o incluso por una reducción del throughput existente; esta situación se presenta cuando múltiples puertos de entrada compiten por el mismo puerto de salida y saturan su ancho de banda. Si la red está congestionada, la utilización es usualmente muy alta, y los paquetes son descartados debido a que la memoria intermedia del dispositivo (buffer) está saturada, y las tasas de colisión son elevadas.

El problema fundamental que genera esta falla es que todos los recursos de red están limitados, incluyendo el tiempo de procesamiento del dispositivo de red (Ej. Router, Switch) y el throughput de cada enlace. Los usuarios pueden sobrecargar fácilmente ciertos recursos de red (de manera similar a un ataque de negación de servicio), inutilizando a la red, a menos que se tomen las medidas necesarias para evitar esta situación.

En el esquema de protocolos TCP/IP generalmente se vigilan los errores, pérdidas o retardos de paquetes. La congestión es un aspecto muy importante, ya que es la causa de toda la pérdida de paquetes en redes conmutadas.

- **Network Paging (Paginación de la red).** Se presenta al paginar o trasladar archivos o procesos muy grandes a través de la red, produce un consumo de ancho de banda excesivo que afecta el desempeño general de la red.

- **Babbling Node (Nodo Murmurante).** Es una estación que transmite repetidamente paquetes aleatorios en la red. Esta falla es característica de una tarjeta de red defectuosa o una mala implementación de protocolo.

- **Stalled bridge (Puente Atascado).** Esta falla ocurre cuando un Bridge o Router entra en estado inactivo (down) o detiene el reenvío de paquetes en una u otra dirección. Esta

falla resulta en pérdida de conexiones, y de este modo reduce la actividad en todas las secciones de la red que se relacionan con el dispositivo. En este tipo de falla también se incluye la pérdida de otros componentes de red o redes desconectadas.

- **Runt flood (Inundación de Fragmentos, enanos).** Esta falla ocurre cuando una estación envía paquetes que son menores al límite de 64 bytes definido por el protocolo Ethernet. La transmisión de paquetes fragmentados resulta en la pérdida de ancho de banda de red.

- **Jabbering node (Inundación de Gigantes).** Esta falla ocurre cuando una estación envía paquetes que son mayores al límite de 1518 bytes definido por el protocolo Ethernet. La transmisión de paquetes de gran tamaño resulta en una pérdida extrema de ancho de banda.

- **Fallas de hardware.** Son problemas específicos con el medio de transporte. Existen muchos tipos de fallas de hardware que pueden ocurrir en una red Ethernet, incluyendo un conector defectuoso, y problemas eléctricos.

Las fallas de Hardware no se detectan fácilmente usando los parámetros de desempeño monitorizados usualmente. Estas fallas se manifiestan generalmente en otros parámetros, como son: el espaciamiento ínter-paquetes, paquetes erróneos, etc.

En general la incidencia de estas fallas es muy reducida en las redes Ethernet bien configuradas. Los problemas de Hardware y también otras fallas como los Runts y Jabbers, se pueden apreciar principalmente como anomalías observables en gráficas de paquetes versus carga y paquetes versus colisiones.

2. SELECCIÓN DE FALLAS Y ASOCIACIÓN CON VARIABLES DE LA MIB

En este capítulo se describirá el proceso y los criterios de selección de las fallas de red Ethernet típicas utilizadas durante este estudio. Además se expondrá la asociación de tales fallas con un conjunto de variables MIB con base en la afinidad entre las características de las fallas y la información almacenada en tales variables. El objetivo es establecer una base conceptual que permita identificar los posibles estados de falla propios de una red a partir de la información obtenida de los dispositivos administrables asociados a ella.

2.1 ANTECEDENTES

En la literatura relacionada con este tema, se presentan con frecuencia procedimientos de selección de variables basados en la experiencia de los investigadores y administradores de las redes bajo estudio. Debido a que el propósito de este trabajo es crear la base para la realización de una metodología general, este tipo de criterios no se ajusta a los objetivos perseguidos.

Las distintas investigaciones realizadas en este tema tienen como objetivo común, el análisis de grandes volúmenes de datos para reducirlos a sólo aquellos eventos anómalos que sean indicativos de problemas en la red. Habitualmente se busca identificar parámetros de desempeño que son afectados por una falla o anomalía específica y luego seleccionar las variables que brinden información detallada sobre estos parámetros y de esta manera establecer objetos de monitorización específicos para cada evento anómalo o falla que se presente en la red.

Además, los diferentes estudios[36] presentan un tratamiento empírico mediante prueba y error en la etapa de selección de las variables, ya que comúnmente dentro del proceso de

[36] Para mayor información revisar en la Bibliografía, THOTTAN, M; JI, Chuanyi.; HOOD, Cynthia S; FEATHER, Frank; SIEWIOREK, Dan; MAXION, Roy ; DUARTE ; Elías Procopio, DOS SANTOS, Aldri L; Et al.

creación de herramientas automatizadas de gestión esta etapa de selección es trivializada por la experiencia de los investigadores.

2.2 CRITERIOS DE SELECCIÓN DE FALLAS

Como se mencionó en el capítulo anterior[37], existe un conjunto de fallas que son muy comunes en redes Ethernet y que han sido estudiadas y documentadas ampliamente. Entre este conjunto tan amplio, es preciso seleccionar un número mínimo de fallas representativas que puedan ser descritas de manera simple a partir de la información almacenada en las variables de la MIB, de tal forma que puedan identificarse los estados de falla mediante la selección adecuada de un grupo de variables asociados a cada uno de ellos.

En general para la selección de las fallas a estudiar en una red Ethernet se deben atender los siguientes criterios:

- La falla debe ocurrir con cierta frecuencia en la red bajo estudio. (las fallas de baja ocurrencia como los Runts y Jabbers generalmente son inyectadas en el sistema periódicamente con el fin de realizar pruebas)

- Estas fallas deben representar un amplio rango de problemas en una red, como el mal uso de la red originado en los protocolos de alto nivel, problemas de implementación de protocolos, tráfico excesivo, saturación de la capacidad de la red, etc.

- La selección de fallas está limitada por los parámetros de desempeño soportados y reportados por el sistema de monitorización disponible.

2.2.1 Procedimiento de selección. Siguiendo los criterios planteados, se procedió a realizar un análisis de las características y propiedades de cada una de las fallas anteriormente mencionadas. Esta valoración reveló que varias de estas fallas cumplen

[37] Ver sección 1.6.2 Tipos de Fallas

ampliamente con los criterios de selección, por lo tanto se decidió tener en cuenta sólo aquellas fallas que representarán de manera general a una variedad de posibles problemas en un segmento de red; además se tuvo preferencia por las fallas que permitieran una fácil implementación en un ambiente de laboratorio para la realización de las pruebas de validación de resultados.

Las fallas seleccionadas según este procedimiento son las de Congestión y la Tormenta Broadcast, ya que cumplen totalmente con los criterios planteados y existe una gran cantidad de variables de la MIB estándar cuya información puede ser relacionada con la presencia de estas fallas en un segmento de red Ethernet, lo que nos provee de una amplia base para la elección de variables. Además su generalidad también permite su fácil implementación en el laboratorio, lo que posibilita el análisis y la obtención de resultados para validar este estudio.

2.3 CRITERIOS DE SELECCIÓN DE VARIABLES MIB

La selección de un número reducido de variables de la MIB constituye el primer paso para el desarrollo de un agente de red, pues este proceso permite reducir la complejidad del problema a un nivel más tratable. Esta actividad adquiere importancia, porque un gran número de variables de la MIB son redundantes en términos de la detección de fallas.

Mediante la comparación entre las características de las fallas seleccionadas y la descripción de las variables de la MIB estándar es posible establecer un proceso de selección de variables que brinden información crítica para la identificación de las condiciones o síntomas asociados a estas fallas en el segmento de red bajo estudio.

Existen dos procedimientos claves[38], muy usados a la hora de la selección adecuada de las variables:

[38] CABRERA, J; LEWIS, L; QIN, X; LEE, W; PRASANTH, R; RAVICHANDRAN, B; MEHRA, R. "Proactive Detection of Distributed Denial of Service Attacks using MIB Traffic Variables a Feasibility Study" IEEE 2001.

- El primero, consiste en usar el conocimiento y experiencia que se tiene acerca de las fallas, sus causas y manifestaciones.

- El segundo consiste en comparar el comportamiento de cada variable durante una falla, con el comportamiento de la variable durante la operación normal.

La primera opción es la clave de este trabajo, ya que la base a utilizar es la definición teórica, la estructura y las propiedades de las fallas de red más comunes dentro de un segmento de red Ethernet.

La segunda opción implica un proceso de monitorización de red que se implementó en este estudio como método de comprobación y criterio final de selección de las variables clave; ya que usar este procedimiento como aproximación inicial conlleva el efectuar una discriminación a priori sin considerar las características teóricas y su posible asociación a las fallas tomadas como referencia.

Dentro de este proceso de monitorización se debe vigilar el comportamiento de los objetos MIB seleccionados. Aquellos que presenten una variabilidad más significativa serán seleccionados finalmente.

Para llegar a la implementación de este segundo procedimiento, con base en las definiciones estándar se debe realizar una preselección de las variables que ofrecen mayor información y permiten su asociación precisa a cada una de las fallas estudiadas.

Los criterios básicos para el proceso de selección de variables de las MIB SNMP son:

- El criterio primordial, usado en este estudio para la selección de las variables MIB, es seleccionar variables cuya definición estándar permita relacionarlas de manera directa con las causas de las fallas bajo estudio, y que igualmente aporten información crítica para la gestión, especialmente aquellas que suministren detalles sobre el estado y el desempeño de

la red.

- El siguiente criterio radica en la selección de variables que hagan parte de grupos estándar de la MIB. Este razonamiento es aplicado ya que en muchos casos los grupos de variables privados, son limitados a los dispositivos del fabricante para los que fueron especificados, además las adiciones del SNMPV2 y RMON2 [39] conocidas como MIB II no han sido normalizadas en los equipos de todos los fabricantes, por esta razón basarse en los grupos MIB estándar, permite una base de selección universal.

- La selección de variables asociadas a los protocolos de uso más frecuente dentro de la red, está justificada como una forma de simplificar la cantidad de información disponible y ajusta el estudio al contexto de las características y propiedades de la red bajo estudio.

- El protocolo SNMP en sus distintas versiones soporta varias clases o sintaxis de objetos, usados para representar información diferente[40]. Al gestionar el estado y funcionalidad de la red es necesario manejar las variables de diversas sintaxis [41]que portan información numérica, en un formato adecuado para la monitorización y su posterior análisis; siguiendo esta premisa se puede concluir que las variables más apropiadas para cumplir estos requerimientos son las que pertenecen a los siguientes tipos de sintaxis: INTEGER, COUNTER, GAUGE y TIME TICKS; Esto se debe a que están completamente determinados por los esquemas dados en la ASN.1[42].

Variables con otros tipos de sintaxis como Octet String, Object Identifier, etc., presentan grandes dificultades a la hora de obtener información cuantitativa a partir de ellas.

---

[39] RFC 1213 y 1157.

[40] SAYENKO, Oleksandr. "Policy Based Model for Monitoring SNMP Resources" Master's Thesis Work Mobile Computing 8/8/2002.

[41] Para mayor información ver el Anexo B.

[42] Abstract Syntax Notation One: Sintaxis Abstracta de Notación Uno. Lenguaje OSI para describir tipos de datos independientes de las estructuras de computadores específicos y técnicas de representación. Descrito en el estándar Internacional de ISO 8824.

Esta característica de las variables u objetos que conforman la MIB, permite establecer un criterio de clasificación. Plantear una metodología general es la orientación de este trabajo, por eso la capacidad de presentar y manejar la información en forma adecuada y simple, se convierte en un factor crítico; por lo tanto el tipo de sintaxis al que pertenecen puede ser usado como principio de discriminación y selección de las variables bajo estudio.

- Elegir los objetos que proporcionen la información de la forma más general posible para descartar las variables que la repitan de manera total o parcial y así evitar la redundancia de información entre variables, que debido al gran número de objetos existentes en la MIB se convierte en un problema frecuente dentro de las tareas de gestión de redes.

Siguiendo estos conceptos es posible iniciar el procedimiento de preselección a partir de las tablas de variables MIB[43] y las fallas de red Ethernet seleccionadas en la sección anterior.

2.3.1 Procedimiento de selección. Después de fijar los criterios base, hay que tener en cuenta que este proceso de selección está orientado a la búsqueda de las variables de la MIB que permitan obtener información clave sobre los síntomas que preceden y describen la presencia de las fallas seleccionadas en un segmento de red específico.

Ya que las fallas seleccionadas son la de congestión y la tormenta broadcast, las variables a elegir serán las que almacenen mayor información sobre aumentos exagerados en la carga de tráfico en el segmento de red y el uso indiscriminado de la transmisión de paquetes de difusión respectivamente.

Ya que en los grupos estándar de la MIB existen gran cantidad de variables que pueden ser fácilmente asociadas a los escenarios de estas fallas, el análisis de las definiciones de cada una de estas variables se hizo de forma estricta, con el fin de escoger solo las variables que aportaran información significativa sobre las fallas y para evitar la redundancia de

[43] Para ver ejemplos de tablas de variables MIB completos revisar el anexo A.

información que se convierte en un problema que surge con frecuencia al intentar clasificar un grupo de variables de la MIB.

Esta situación se produce por la existencia de variables diferentes con definiciones similares, pero que en el momento de realizar el proceso de monitorización pueden mostrar valores significativamente diferentes, lo que generaría confusión al recopilar los datos almacenados en las variables, y no permitiría validar los resultados que deben confirmar su asociación con los escenarios de falla seleccionados con anterioridad.

Al realizar la primera selección de variables se obtuvieron alrededor de 77 con las propiedades adecuadas[44], después de un análisis exhaustivo de sus propiedades y teniendo en cuenta que las variables con información de mayor importancia asociada a las fallas bajo estudio son las pertenecientes a los grupos de variables representativos de protocolos de capas altas y los que están asociados directamente con las interfaces de red que hacen parte del segmento[45], podemos asegurar que entre estos grupos los más importantes son: Interfaces, IP, ICMP, SNMP y RMON; esta premisa condujo a la preselección de variables que nos permitió reducir a sólo 30 el grupo inicial. Estas variables son presentadas para cada falla en las siguientes tablas.

Tabla 1. Grupo inicial de variables de la MIB asociadas a la falla de Tormenta Broadcast

TORMENTA BROADCAST		
NOMBRE	**OID**	**DESCRIPCIÓN**
IfInNUcastPkts	1.3.6.1.2.1.2.2.1.12	Muestra el número de paquetes entregados por esta subcapa a una superior, los cuales fueron direccionados a una dirección Broadcast o multicast en esta subcapa.

[44] Para observar estas Tablas revisar el anexo A.

[45] THOTTAN, M; JI, Chuanyi. "Anomaly Detection in IP Networks". Bell Labs 2002.

IfOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	Muestra el número total de paquetes que los protocolos de capas altas solicitaron para ser transmitidos y que fueron direccionados a una dirección broadcast o multicast en esta subcapa; incluyendo aquellos que fueron descartados o no enviados.
EtherStats BroadcastPkts	1.3.6.1.2.1.16.1.1.1.6	Muestra el número total de paquetes bien formados recibidos que fueron dirigidos a la dirección broadcast. No se incluyen los paquetes multicast.
Rip2GlobalQueries	1.3.6.1.2.1.23.1.2	Indica el número de respuestas enviadas a consultas RIP desde otros sistemas.

Tabla 2. Grupo inicial de variables de la MIB asociadas a la falla de Congestión

CONGESTIÓN		
NOMBRE	**OID**	**DESCRIPCIÓN**
IfInOctets	1.3.6.1.2.1.2.2.1.10	Muestra el número total de octetos recibidos en la interfaz, incluyendo caracteres de trama (frame).
IfInUcastPkts	1.3.6.1.2.1.2.2.1.11	Muestra el número de paquetes entregados por esta subcapa a una superior, los cuales no fueron direccionados a una dirección Broadcast ni multicast en esta subcapa.
IfInDiscards	1.3.6.1.2.1.2.2.1.13	Muestra el número de paquetes entrantes que fueron elegidos para ser descartados aunque no se hayan detectado errores, para evitar que sean entregados a un protocolo de capa superior. Una posible razón para descartar tales paquetes es la liberación de espacio en los buffer.

IfInErrors	1.3.6.1.2.1.2.2.1.14	Para interfaces orientadas a paquetes, muestra el número de paquetes de entrada que contienen errores, evitando su entrega a protocolos de capa superior. Para interfaces orientadas a caracteres o de longitud fija, muestra el número de unidades de transmisión de entrada que contienen errores, evitando su entrega a protocolos de capa superior.
IfOutOctets	1.3.6.1.2.1.2.2.1.16	Muestra el número total de octetos transmitidos fuera de la interfaz incluyendo caracteres de tramas.
IfOutUcastPkts	1.3.6.1.2.1.2.2.1.17	Muestra el número total de paquetes que los protocolos de capas altas solicitaron para ser transmitidos y que no fueron direccionados a una dirección broadcast o multicast en esta subcapa; incluyendo aquellos que fueron descartados o no enviados.
IfOutDiscards	1.3.6.1.2.1.2.2.1.19	Muestra el número de paquetes de salida que fueron elegidos para ser descartados aunque no se hayan detectado errores, para evitar que sean transmitidos. Una posible razón para descartar tales paquetes es la liberación de espacio en los buffer.
IfOutQLen	1.3.6.1.2.1.2.2.1.21	Muestra la longitud de la cola de paquetes de salida (en paquetes).
IpInReceives	1.3.6.1.2.1.4.3	Indica el número total de datagramas de entrada recibidos desde las interfaces, incluyendo aquellos recibidos con errores.
IpForwDatagrams	1.3.6.1.2.1.4.6	Indica el número de datagramas de entrada para los que esta entidad no fue su destino IP final, como resultado de esto se hizo el intento de encontrar una ruta para reenviarlos a su destino final. En entidades que no actúan

		como routers IP, este contador incluirá solo aquellos paquetes que fueron enrutados desde el origen a través de esta entidad, y el procesamiento de la opción de enrutamiento desde el origen fue exitoso.
IpInDiscards	1.3.6.1.2.1.4.8	Indica el número de datagramas IP de entrada para los que no hubo problemas al prevenir su procesamiento constante, pero que fueron descartados Este contador no incluye ningún datagrama descartado mientras espera su reensamble.
IpInDelivers	1.3.6.1.2.1.4.9	Indica el número total de datagramas de entrada entregados satisfactoriamente a los protocolos IP de usuario (incluyendo ICMP).
IpOutRequest	1.3.6.1.2.1.4.10	Indica el número total de datagramas cuyos protocolos IP de usuario local (incluyendo ICMP) suplieron las peticiones de IP para transmisión. Este contador no incluye ninguna datagrama contado en ipForwDatagrams.
IpOutDiscards	1.3.6.1.2.1.4.11	Indica el número de datagramas de entrada para los que no hubo problemas al prevenir su transmisión a su destino, pero que fueron descartados (por falta de espacio en los buffer).
IpFragCreates	1.3.6.1.2.1.4.19	Indica el número de fragmentos de datagramas IP que han sido generados como resultado de la fragmentación en esta entidad.
IpRoutingDiscards	1.3.6.1.2.1.4.23	Muestra el número de entradas de enrutamiento que fueron escogidas para ser descartadas aun siendo validas. Una razón para descartar tales entradas podría ser para liberar espacio en los buffer para otras entradas de

		enrutamiento.
IcmpInSrc Quenchs	1.3.6.1.2.1.5.6	Indica el número de mensajes ICMP de fuente desconectada recibidos
IcmpOutSrc Quenchs	1.3.6.1.2.1.5.19	Indica el número de mensajes ICMP de fuente desconectada enviados.
TcpInSegs	1.3.6.1.2.1.6.10	Indica el número total de segmentos recibidos, incluyendo aquellos recibidos con errores. Esta cuenta incluye los segmentos recibidos en las conexiones establecidas actualmente.
SnmpInPkts	1.3.6.1.2.1.11.1	Muestra el número total de mensajes entregados a una entidad SNMP desde el servicio de transporte.
SnmpOutPkts	1.3.6.1.2.1.11.2	Muestra el número total de mensajes SNMP que fueron aprobados desde una entidad de protocolo SNMP hasta el servicio de transporte.
SnmpInTraps	1.3.6.1.2.1.11.19	Muestra el número total de PDUs Traps SNMP que han sido aceptadas y procesadas por la entidad de protocolo SNMP.
SnmpOutTraps	1.3.6.1.2.1.11.29	Muestra el número total de PDUs Traps SNMP que han sido generadas por la entidad de protocolo SNMP.
EtherStatsOctets	1.3.6.1.2.1.16.1.1.1.4	Muestra el número total de octetos de datos (incluyendo aquellos en paquetes defectuosos) recibidos en la red (excluyendo los bits de entramado pero incluyendo octetos FCS). Esta variable puede ser usada como un estimado razonable de la utilización de Ethernet.

EtherStatsPkts	1.3.6.1.2.1.16.1.1.1.5	Muestra el número total de paquetes (incluyendo los paquetes defectuosos, paquetes de broadcast, y paquetes multicast) recibidos.
EtherStats Collisions	1.3.6.1.2.1.16.1.1.1.13	Indica el número total de colisiones mejor estimado (más preciso) en este segmento Ethernet. El valor retornado dependerá de la localización del sondeo RMON.

Debido a que la metodología que se quiere establecer debe estar orientada a la creación de herramientas, que realicen las tareas de gestión de redes de manera automatizada, apoyándose en la información almacenada en las variables de la MIB, se debe tener especial cuidado con la cantidad de variables a seleccionar. Como la cantidad de variables que manejen este tipo de aplicaciones de gestión estará directamente relacionada con su desempeño y consumo de recursos de hardware y de ancho de banda en la red, se debe seleccionar una cantidad mínima de variables asociadas a cada falla.

De esta manera la cantidad de variables que hemos seleccionado a partir de los criterios teóricos que han sido establecidos es aún demasiado alta, y para reducir su número se deben aplicar parámetros particulares definidos por los recursos con los que se cuenta y los esquemas de trabajo planteados en el proceso de investigación, como por ejemplo las herramientas software a utilizar en las etapas de pruebas y los esquemas elegidos para realizarlas. En los próximos capítulos se expondrán los procesos y parámetros complementarios aplicados sucesivamente durante cada una de las etapas de este estudio para minimizar progresivamente el número de variables de la MIB asociadas a cada falla, y así facilitar el establecimiento de una metodología general para este tipo de procedimiento.

3. EVALUACIÓN Y SELECCIÓN DE HERRAMIENTAS SOFTWARE

Dentro de las actividades a realizar, las pruebas de laboratorio se convierten en la herramienta principal a la hora de confirmar los planteamientos teóricos hechos inicialmente. Como parte de estas pruebas de laboratorio, las cuales se tratarán en el siguiente capítulo, la selección y aplicación de las herramientas software para la obtención de los valores de las variables para su posterior análisis y la evaluación y selección de las herramientas de generación de tráfico adecuadas que nos permitan crear un escenario de falla, son un paso fundamental.

Este proceso comenzó con la identificación y obtención de las herramientas; se realizó una búsqueda en Internet, que permitió encontrar diferentes opciones con las cuales iniciar la tarea de selección.

3.1 EVALUACIÓN Y SELECCIÓN DE HERRAMIENTAS SOFTWARE DE MONITORIZACIÓN

Un Monitor de Tráfico en Redes de Computadores es un instrumento que entrega datos acerca de la red en la cual está conectado. Estos datos pueden ser entregados en diversas formas, dependiendo del fin con el cual el monitor es diseñado.

Para este trabajo, la herramienta de monitorización debe estar basada en el protocolo SNMP para que proporcione la información almacenada en las variables de la MIB.[46]

Para la selección del software se tienen en cuenta los siguientes criterios:

[46] En el anexo C se explican las características para un tipo de monitorización basada en el protocolo SNMP.

1. La herramienta software debe incluir en su base de datos, como mínimo los módulos básicos del SMI y los grupos estándar de la MIB mencionados en el capítulo 2.
2. Debe permitir, establecer la frecuencia de sondeo de las variables de la MIB, encuestar distintas variables simultáneamente; visualizar el sondeo de manera gráfica; y además, tener la capacidad de almacenar los datos en un formato universal para su posterior manipulación y análisis.

3. Como sistema de gestión, también se deben tener en cuenta aquellas herramientas adicionales que el software seleccionado ofrezca y que puedan ayudar al administrador de la red.

4. Se preferirán las herramientas con un entorno gráfico agradable y de fácil manejo para el usuario.

Luego de identificar estos criterios, se seleccionaron las siguientes herramientas para las cuales se presenta una breve descripción[47]:

3.1.1  MG-SOFT MIB BROWSER (Edición profesional).

Versión 9.0.0.3991 1995-2003 Corporación MG-SOFT.

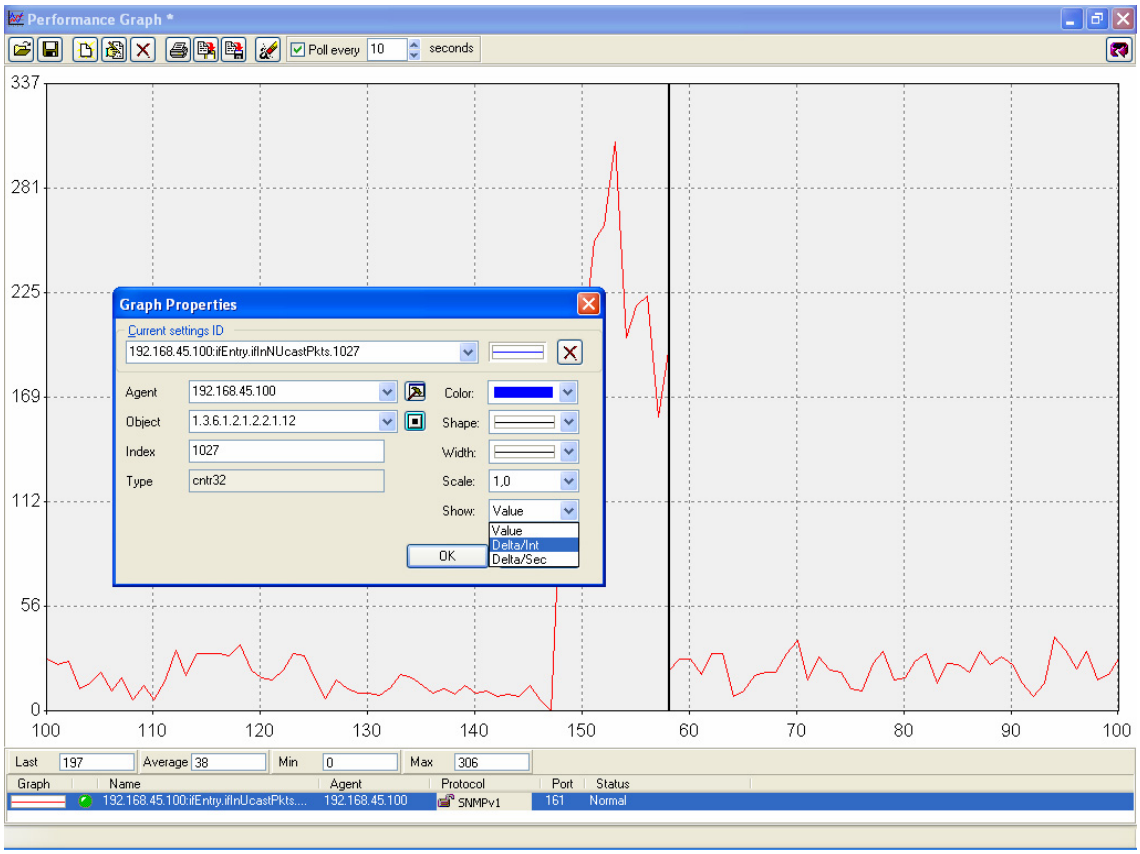
Es una herramienta que permite monitorizar y administrar cualquier dispositivo de red gestionable mediante el uso de cualquiera de las versiones del protocolo SNMP; esta herramienta se especializa en la construcción y organización de estructuras de información de gestión SMI[48], permite al usuario crear, comparar y compilar módulos MIB propios. Esta compuesta por las herramientas MIB BROWSER, MIB COMPILER, MIB BUILDER y MIB EXPLORER.

[47] En el anexo D se presenta información más detallada de cada software de monitorización y cada una de las herramientas adicionales que son de interés en este estudio.

[48] SMI = Estructura de información de gestión que define un lenguaje formal para la descripción de la información de gestión de modo que la información sea recuperable y modificable.

Se utilizó una versión de evaluación descargada de http://www.mg-soft.com/ que tiene algunas limitaciones; sólo permite trabajar sobre el protocolo SNMPV1 y los 10 grupos estándar (RFC1213) y no se tiene acceso a los módulos SNMPV2, SNMPV3 y RMON.

Figura 6. Herramienta gráfica de MG-SOFT.



3.1.2 SOLARWINDS Professional Edition 5.2.

Versión 5.0.58 1995-2002 Corporación Solarwinds.

Este aplicativo está compuesto por varias herramientas dedicadas a la administración, la monitorización y al descubrimiento de la red.

Dentro de sus 9 paquetes de herramientas, aquellas que brindan información sobre las

variables de la MIB y que pueden servir para la monitorización de éstas, son: el MIB BROWSER y el PERFORMANCE MONITORING.

- **MIB browser**. Contiene un navegador completo de la MIB, que permite explorar el árbol, encuestar la información de cualquier OID y modificar remotamente valores de SNMP.

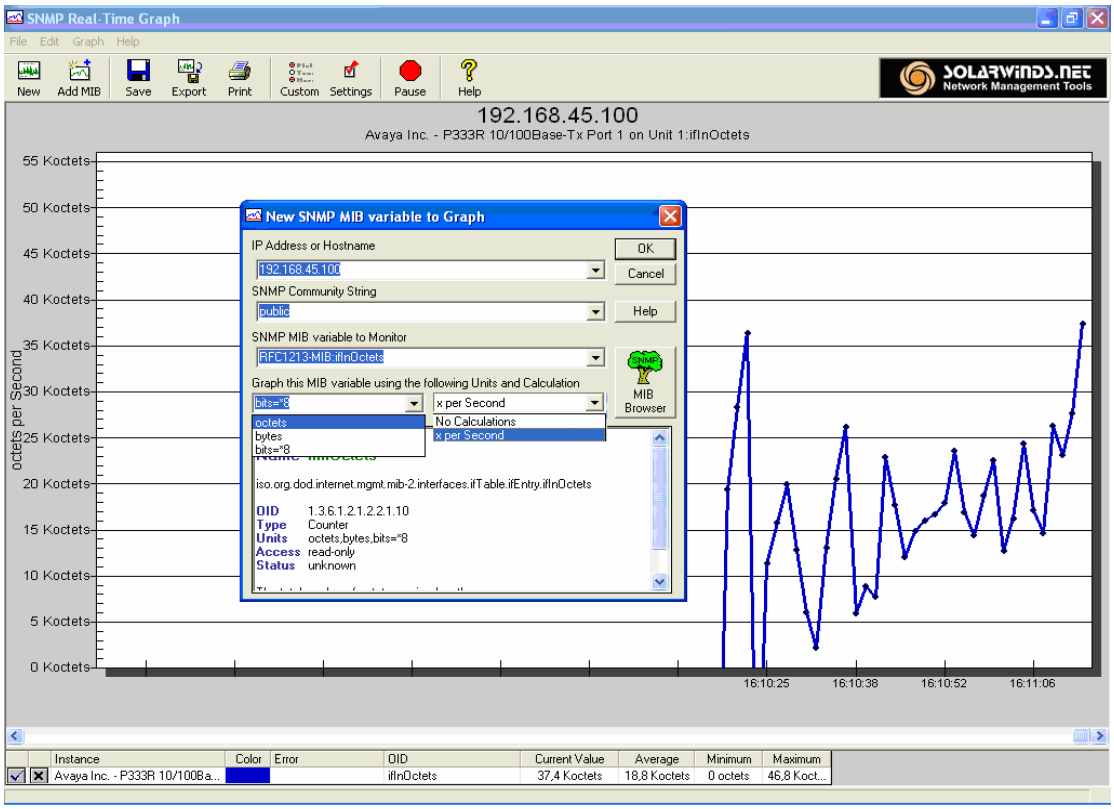
La base de datos de SolarWinds contiene más de 1.000 módulos MIB y más de 100.000 OIDS públicos y privados (de fabricante).

SolarWinds, incluye otras herramientas que realizan funciones basadas en el protocolo SNMP tales como: MIB Walk, el cual genera una tabla de todas las MIB y OIDS soportadas en un dispositivo específico, Update System MIB, y MIB Viewer.

- **Performance monitoring.** Consta de un conjunto de herramientas que permiten monitorizar el desempeño de dispositivos mediante cálculos basados en información estadística suministrada por las variables de la MIB; en la mayoría de herramientas la información de las variables es transparente para el usuario.

Dentro de este grupo, la herramienta SNMP-Graph permite monitorizar las variables MIB en tiempo real, muestra gráficamente datos de cualquier variable simplemente seleccionando el dispositivo y el OID deseado.

Figura 7. Herramienta gráfica de SOLARWINDS SNMP-Graph.



En la tabla 3 se presenta una comparación de las características más sobresalientes que resumen las utilidades y las limitaciones de las diferentes herramientas de monitorización analizadas.

Se decidió seleccionar el software SOLARWINDS como la herramienta para llevar a cabo la monitorización de las variables, por su agradable presentación y facilidad de configuración para la monitorización y el registro de los datos; además permite acceder fácilmente a cualquier herramienta complementaria a través de enlaces en la mayoría de sus aplicaciones.

Tabla 3. Comparación de herramientas de monitorización

CARACTERÍSTICAS			MG-SOFT MIB BROWSER	SOLARWINDS SNMP-GRAPH MIB BROWSER
SISTEMA OPERATIVO			WINDOWS, LINUX	WINDOWS
PROTOCOLOS SNMP SOPORTADOS			SNMPV1	SNMPV1/V2/V3
PERMITE NAVEGAR EL ÁRBOL DE LA MIB			SI	SI
COMMANDOS SNMP SOPORTADOS	GET		SI	SI
	GETNEXT		SI	SI
	GETBULK		SI	SI
	SET		SI	SI
GUARDA INFORMACIÓN EN ESTOS TIPOS DE ARCHIVOS			CSV, TXT	XLS, CSV, TXT
CONFIGURACION DE TRAPS E INFORMES			SI	SI
MUESTRA EL MAPA DE LA RED			-	-
MUESTRA Y ORGANIZA TABLAS SNMP			SI	SI
GRAFICA LOS VALORES DEL OID EN TIEMPO REAL			SI "PERFORMANCE GRAPH"	SI "SNMP GRAPH"
ICMP PING			SI	SI
NOTIFICACIONES VÍA E-MAIL			-	SI
REPORTES VÍA WEB			-	SI
MODULOS MIB	10 GRUPOS ESTÁNDAR BÁSICOS		SI	SI
	BASE DE DATOS PARA OIDS PRECOMPILADOS		SE CARGAN A TRAVÉS DE MG-SOFT INC.	MAS DE 100.000 OIDS
	GRUPO RMON		NO EN LA VERSIÓN DE EVALUACIÓN.	SI
	GRUPOS FABRICANTES		SE CARGAN A TRAVÉS DE MG-SOFT INC.	MÁS DE 1000 MÓDULOS
RANGO DE FRECUENCIA DE SONDEO			DE 1 A 7200Seg	DE 1 A 600Seg
HERRAMIENTA GRÁFICA	GRAFICA DISTINTAS VARIABLES SIMULTÁNEAMENTE		SI	SI
	FIJA UNIDADES PARA LAS VARIABLES		-	SI
	OBTENCIÓN DE LOS DATOS	VALOR ACUMULATIVO (SIN CALCULOS)	SI	SI
		UNIDADES / SEGUNDO	SI	SI
		UNIDADES / PERIODO DE MUESTREO	SI	-

3.2. EVALUACIÓN Y SELECCIÓN DE HERRAMIENTAS SOFTWARE DE GENERACIÓN DE TRÁFICO.

Después de obtener algunas herramientas de generación representativas, las cuales son totalmente gratis o en la modalidad de versión de evaluación por 30 días, se aplicaron los siguientes criterios para la selección de la herramienta a utilizar:

1. Capacidad y utilidad de estas aplicaciones para generar las fallas objeto de este estudio.

2. La principal característica de funcionamiento que deben ofrecer las herramientas a elegir, es la capacidad de generar diferentes patrones de tráfico y que permitan modificar parámetros como la carga total a generar, el tamaño y distancia entre paquetes, manejar los tiempos de conexión y elegir los puertos para cada una de ellas, además que posean la capacidad de guardar los datos del proceso de generación en formatos que luego puedan ser visualizados fácilmente para su posterior análisis.

3. Se prefirieron herramientas con un entorno gráfico agradable y de fácil manejo para el usuario.

Después de evaluar las anteriores características en las herramientas, se seleccionaron aquellas que funcionan en plataformas Windows XP, ya que esto brinda mayor compatibilidad con todas las aplicaciones disponibles en el laboratorio y con las herramientas de monitorización seleccionadas.

Al terminar este proceso de selección se determinó utilizar las siguientes herramientas asociadas a cada escenario de falla[49].

[49] En el anexo D se presenta información detallada para cada herramienta de generación asociada al respectivo escenario de falla.

3.2.1. Herramientas software seleccionadas para la falla de congestión.

- **Wan killer.** Este programa hace parte del paquete "Miscellaneous" de SolarWinds Professional Edition Ver 5.0.58 Demo por 30 días.

Es una herramienta de generación de tráfico, que permite definir el porcentaje de ancho de banda del canal a ocupar con el tráfico generado, también permite ajustar el tamaño del paquete generado. Para su utilización se debe seleccionar el protocolo de transporte (TCP o UDP) y el número del puerto.

- **Lan Traffic V2.** LanTraffic V2 permite la generación de tráfico UDP y TCP en una red IP. Puede ser configurado con un gran número de parámetros diferentes y soporta múltiples conexiones IP simultáneas. Esta herramienta comprende dos módulos: el emisor y el receptor, lo que hace posible medir el RTT (Round Trip Time) en cada conexión; gracias a ello LanTraffic puede ser usado en mediciones de desempeño de redes IP.

- **TFGen V1.0.** Permite generar tráfico en la red de área local (LAN), está diseñado para trabajar bajo redes TCP/IP únicamente. Utiliza el protocolo de transporte UDP por lo que requiere al menos un nodo IP de destino. Tiene la capacidad de generar tráfico Multicast y puede generar tráfico con patrones específicos.
Esta herramienta es freeware[50] y su instalación está libre de spyware[51]

3.2.2 Herramientas software seleccionadas para la falla de tormenta de broadcast.

- **TFGen V1.0.** Como se describió anteriormente, esta herramienta utiliza el protocolo de transporte UDP por lo que requiere al menos un nodo IP de destino, luego permite que se configure para generar broadcast IP a las estaciones conectadas al segmento

[50] Freeware es un software que se distribuye sin costo alguno.

[51] Spyware se denomina a los archivos o aplicaciones de software que son instalados en los sistemas, algunas veces sin conocimiento u autorización de los usuarios, generalmente acompañan a las aplicaciones gratuitas.

de red bajo estudio.

- **Formas complementarias de generación.** Otra forma de generar tráfico broadcast, pero de tipo ARP es utilizar aquellas herramientas que realicen funciones de descubrimiento de la red.

Dentro de DISCOVERY NETWORK de SolarWinds existen varias herramientas de descubrimiento de la red, que indirectamente se pueden utilizar para la generación de tráfico broadcast ARP, al ejecutar cualquiera de estas herramientas tales como: **IP Network Browser, Ping Sweep, SNMP Sweep** y **MAC Address Discovery.**

Estas herramientas funcionan realizando un barrido en la subred que ocasiona la consulta de cada dirección IP, determinando cuales son las estaciones que están conectadas, y sus respectivos nombres de dominio o direcciones MAC, además informa si estas estaciones soportan el protocolo SNMP.

Para realizar estas tareas la aplicación produce una gran cantidad de tráfico broadcast que podemos aprovechar para generar este tipo falla.

En la tabla 4 se presenta una comparación de las características más sobresalientes que resumen las utilidades y las limitaciones de las diferentes herramientas de generación analizadas.

Tabla 4. Comparación de herramientas de generación de tráfico

CARACTERÍSTICAS		WAN KILLER	TFGEN V1.0	LANTRAFFIC V2
TIPO DE PAQUETE		UDP, TCP	UDP	UDP, TCP
DEFINICIÓN DE PUERTO DE SALIDA		SI	SI	SI
DEFINICIÓN DE PUERTO DE ENTRADA		-	-	SI
DEFINICIÓN DE ANCHO DE BANDA		SI	SI	SI
DEFINICIÓN DE TAMAÑO DE PAQUETES	CONSTANTE	SI, % DE ANCHO DE BANDA	SI, % DE ANCHO DE BANDA	SI
	ALEATORIO	-	-	SI
	ALTERNADO	-	-	SI
	INCREM / DECREMENTAL	-	-	SI
PATRÓN DE TRÁFICO	CONSTANTE	SI, % DE ANCHO DE BANDA	SI, % DE ANCHO DE BANDA	SI
	LEY MATEMATICA	-	-	SI
	ARCHIVO	-	-	SI
	ALEATORIO	-	SI	SI
	PERIODICO	-	SI	SI
RETARDO INTER-PAQUETE	CONSTANTE	SI	SI	SI
	ALEATORIO	-	-	SI
	ALTERNADO	-	-	SI
	INCREM / DECREMENTAL	-	-	SI
MODO DE RECEPCIÓN		ECHO, DISCARD	ECHO	ECHO, ABSORBER
MULTIPLES CONEXIONES (IP DESTINOS)		-	-	SI
GENERA TRÁFICO A DIRECCIÓN MULTICAST (PUEDE UTILIZARSE PARA GENERAR BROADCAST)		-	SI	-
ESTADÍSTICAS Y PARÁMETROS DE TRÁFICO (THROUGHTPUT, DATOS, ERRORES).		-	-	SI

4. ESQUEMAS DE PRUEBA

En este capítulo se presentarán los esquemas y las configuraciones de las herramientas seleccionadas,[52] que serán utilizadas para la implementación de los escenarios de falla propuestos y de esta manera obtener información que facilite la validación de las relaciones entre las variables de la MIB y las condiciones de falla seleccionadas.

4.1 ANTECEDENTES

En la literatura existente sobre este tema es común observar dos tendencias muy marcadas en la definición de esquemas de prueba para la validación de las hipótesis planteadas. Estos procedimientos se sustentan en la creación de esquemas basados en inteligencia artificial o en algoritmos de detección simple que toman ventaja de las propiedades de las variables de la MIB.

Infortunadamente la mayoría de esquemas basados en inteligencia artificial presentan el inconveniente de ser dependientes de un conocimiento previo muy amplio sobre las condiciones de falla en la red bajo estudio y las reglas desarrolladas no se adaptan bien a un ambiente de red variable, debido a que son muy específicas y dependen en gran manera de la experiencia del administrador de la red[53].

Los esquemas basados en algoritmos de detección simple aprovechan las propiedades estadísticas de las variables de la MIB mediante esquemas de combinación simple y son antecedidos por la elección de un grupo óptimo de variables MIB para su estudio.

---

[52] Para mayor información sobre las herramientas seleccionadas ver el capítulo 3.

[53] THOTTAN, M; JI, Chuanyi. "Adaptive Thresholding for Proactive Network Problem Detection" Rensselaer Polytechnic Institute 1999.

La configuración típica de los esquemas de prueba inicia con la realización de un proceso de monitorización que permita obtener la información base sobre el segmento de red que se estudia. Esta información es complementada siguiendo diversos planteamientos que permitan visualizar los parámetros y comportamientos que se quieren estudiar con el uso, por ejemplo, de filtros que rastreen alguna característica en especial, mediante la generación de patrones de tráfico definidos, mediante generación de fallas, etc.

4.2 ESQUEMA ELEGIDO

En el desarrollo de las pruebas asociadas a este trabajo se pueden diferenciar dos grandes actividades: la monitorización de las variables de la MIB y la generación de tráfico que permita implementar los escenarios de falla previamente seleccionados.

4.2.1 Monitorización. Para obtener información del comportamiento del segmento bajo estudio se realizó un sondeo de las variables de la MIB seleccionadas previamente, en los dispositivos gestionables que hacen parte del segmento sobre el que se trabajó, utilizando la herramienta SNMP Graph[54].

Para la realización de esta tarea existen parámetros que deben ser definidos con anticipación y que deben ajustarse al tipo de información que se quiere obtener. El más importante de estos factores es la frecuencia con la que se sondean los dispositivos, la cual esta limitada generalmente[55] por la capacidad de almacenamiento de datos del equipo que realiza el sondeo y por el impacto que este proceso genere sobre la red.

Al seleccionar esta frecuencia de sondeo autores como Stallings[56] recomiendan intervalos muy cortos con el fin de detectar y responder adecuadamente a las fallas que se presenten

[54] Ver capítulo 3.

[55] THOTTAN, M; JI, Chuanyi. "Statistical Detection of Enterprise Network Problems". Rensselaer Polytechnic Institute 1999.

[56] STALLINGS William. "SNMP, SNMPV2, SNMPV3, AND RMON1 AND 2".

en la red, pero obtener esta información a partir de los dispositivos gestionables con una frecuencia muy alta puede convertirse en un problema debido al gran incremento de tráfico y al consumo de recursos del dispositivo que producen las operaciones de sondeo SNMP[57].

Debido a que no existe un consenso sobre este aspecto en la literatura existente y teniendo en cuenta las limitaciones mencionadas, el tiempo promedio de sondeo en una red de tamaño mediano esta alrededor de los 15 minutos[58]; siguiendo estos lineamientos y teniendo en cuenta las condiciones del segmento de red bajo estudio, se decidió realizar el sondeo de los dispositivos gestionables que hacen parte de la prueba con frecuencias de 2, 5 y 10 minutos para obtener un volumen de información suficiente sobre el comportamiento de las variables MIB seleccionadas. De esta forma se asegura la adquisición de datos que permitan establecer un nivel de detección de fallas adecuado.

- **Implementación de la monitorización.** Después de definida la frecuencia de sondeo, se deben establecer el número y tipo de campañas de tomas de datos que se efectuarán y bajo que condiciones, con el fin de limitar la información adquirida mediante el proceso de sondeo, sólo a aquella que resulte útil para cumplir los objetivos del trabajo.

Muchos autores recomiendan realizar una monitorización previa del segmento de red bajo condiciones de operación normales, con el fin de tener un conjunto de datos de referencia que podrían ser útiles al momento de analizar el comportamiento estadístico de las variables MIB, pero la naturaleza dinámica de las redes de datos, conduce a que su desempeño sea variante en el tiempo[59]. Esto indica que la utilización de una red varía con la hora del día, el día de la semana y la temporada del año, y de esta manera cualquier sistema o herramienta que quiera efectuar una detección de condiciones de falla adecuadamente

[57] BREITBART, Yuri; CHAN, Chee-Yong; GAROFALAKIS, Minos; RASTOGI, Rajeev; SILBERSCHATZ, Avi. "Efficiently Monitoring Bandwidth and Latency in IP Networks".

[58] SAYENKO, Oleksandr. "Policy Based Model for Monitoring SNMP Resources" Master's Thesis Work Mobile Computing.

[59] FEATHER, Frank; SIEWIOREK, Dan; MAXION, Roy. "Fault Detection in an Ethernet Network Using Anomaly Signature Matching" SIGCOMM 1993.

debe tener en cuenta la naturaleza variante del tráfico de red y estar en capacidad de adaptarse a estos cambios.

Ya que para identificar las condiciones de falla a partir de las variables MIB se ha seguido el principio de que las fallas de red se manifiestan mediante un comportamiento estadísticamente inusual de los valores de estas variables, se definió que el procedimiento a seguir en este trabajo para determinar esta variación estadística en la MIB consistiría en realizar una monitorización del segmento de red bajo las condiciones especificas de cada uno de los escenarios de falla seleccionados para su estudio y que serán generadas mediante las herramientas de generación de tráfico ya elegidas.

Buscando la adaptación a estas condiciones se decidió realizar la monitorización teniendo en cuenta las frecuencias de sondeo definidas, en el horario laboral, para apreciar la incidencia de las condiciones de falla creadas, y tener una visión clara de cómo se reflejan estos cambios en las variables de la MIB.

4.2.2 Generación de tráfico. Para cumplir con el objetivo de implementar los escenarios de falla seleccionados la segunda actividad a definir es la generación de tráfico.

Esta actividad consiste en la inyección de tráfico en el segmento de red bajo estudio haciendo uso de las herramientas software seleccionadas,[60] y de esta forma se crean las condiciones específicas que se asocian a los escenarios de falla, facilitando la observación de sus efectos en las variables MIB estudiadas.

- **Implementación.** La adecuada realización de esta tarea depende de parámetros que deben ser definidos de tal forma que se puedan emular las condiciones particulares de falla estudiadas con anterioridad[61].

[60] Para mayor información sobre las herramientas seleccionadas ver el capítulo 3.

[61] Ver sección 1.6.

Los parámetros más importantes para esta investigación son: patrón de tráfico, tamaño de paquete, volumen de tráfico generado, tiempo que dura la generación y tipo de paquete.

- **Patrón de tráfico.** Como ya se mostró, las herramientas de generación con las que se cuenta, permiten una gran versatilidad en cuanto al tipo de tráfico a generar. Teniendo en cuenta las condiciones del segmento de red de la prueba y las características de las fallas que se están estudiando se decidió que los patrones de generación más indicados son tráfico continuo y constante, y tráfico continuo y aleatorio, ya que de esta forma se pueden generar las cantidades de tráfico que dan origen a la congestión del segmento de red y al volumen de paquetes de difusión suficiente para crear una tormenta de broadcast.

- **Tamaño de paquete.** Este parámetro puede ser utilizado para regular la cantidad de tráfico generado, ya que a menor tamaño el volumen de paquetes generados será mucho mayor. Esta característica es especialmente útil al momento de generar la falla de congestión, ya que cada uno de estos paquetes origina un paquete de respuesta y el procesamiento de cada una de estas peticiones, produce un consumo exagerado de los recursos de los dispositivos de red, así como la ocupación de los enlaces por la gran cantidad de paquetes que se encuentran circulando en el segmento.

Cuando el tamaño de los paquetes generados es grande, se produce un aumento en el tiempo de respuesta por parte de los dispositivos, debido a que se encuentran ocupados procesando cada uno de estos paquetes para ajustarlos a los tamaños estándar que permitan su transmisión, esto genera grandes retrasos ya que los dispositivos de red ponen estos paquetes en cola mientras son procesados uno por uno.

Teniendo en cuenta estas situaciones y siguiendo los tamaños de paquete establecidos en el estándar de Ethernet[62] se eligieron tamaños de paquete en rangos que están entre 180 y 500 bytes como tamaño pequeño; y mayores a 1518 como tamaño grande.

[62] El tamaño de paquete del estándar Ethernet está entre 64 y 1518 bytes.

- **Volumen de tráfico generado.** Este es el parámetro principal en el que se ha basado el esquema de prueba implementado. Su importancia radica en que las fallas seleccionadas para este estudio están directamente relacionadas con la cantidad de tráfico presente en el segmento de red, y en como es afectado el desempeño de los dispositivos que hacen parte de este. Como se puede advertir el volumen a generar esta relacionado directamente con los dos parámetros anteriores y por lo tanto deben ser establecidos en conjunto y teniendo en cuenta las características del segmento de red en particular. Las herramientas de generación con que se cuenta permiten establecer el volumen de tráfico a generar como un porcentaje de la capacidad de los enlaces que hacen parte del segmento de red.

Ya que el objetivo de estas pruebas es emular condiciones de falla se definió cargar al segmento de red con rangos de volúmenes de tráfico que están entre el 50% y 100%[63] de la capacidad de los enlaces con el objetivo de originar las situaciones que conduzcan a la condición de falla.

- **Duración de la generación.** Este parámetro es importante ya que del tiempo que dure la generación dependerán los efectos que se produzcan en el desempeño del segmento de red y que conllevan a la condición de falla. Además el interés no radica sólo en crear la falla sino en sostenerla el suficiente tiempo para que se vea reflejada en el comportamiento de las variables de la MIB.

Para que la falla pueda visualizarse en un número suficiente de muestras del proceso de monitorización y teniendo en cuenta las frecuencias de sondeo elegidas de 2, 5 y 10 minutos se decidió establecer como duración de la generación de tráfico un tiempo promedio de 15 minutos.

- **Tipo de paquete.** Para implementar los escenarios de las fallas elegidas y hacer que los parámetros de volumen de tráfico y duración de la generación cumplan con su objetivo, es

[63] Estos porcentajes están limitados por la capacidad de las estaciones desde las cuales se genera el tráfico y sus respectivas tarjetas de red. Se puede considerar que estas herramientas funcionan bajo el principio del mejor esfuerzo para generar estos volúmenes de tráfico.

necesario que el tipo de paquete fijado sea de paquetes UDP; pues este tipo de paquete es transmitido sin tener en cuenta el establecimiento de una conexión.

Una vez de establecidos estos parámetros, el paso a seguir fue la configuración de las herramientas de generación de tráfico, para la realización de las pruebas. Esta configuración es presentada para cada falla en las siguientes tablas.

Tabla 5. Configuración de las herramientas para la Falla de Congestión

HERRAMIENTA DE GENERACIÓN DE TRÁFICO	NIVEL DE VOLUMEN DE TRÁFICO GENERADO	PATRÓN DE TRÁFICO	TAMAÑO DE PAQUETE [Bytes]	TIPO DE PAQUETE Y PUERTO	No DE EQUIPOS GENERADORES
WAN KILLER	50%, 70% , 100% [% de BW del canal]	Continuo y constante	180-500	UDP PTO7 (ECHO)	1
			1500-10000	UDP PTO9 DISCARD	
TF-GEN	0, 10, 1000,10000 [Utilización en KBytes]	Continuo y aleatorio	Se auto-ajusta	UDP PTO7 (ECHO)	1
LAN – TRAFFICv2	1-1000 [Utilización en KBytes]	Continuo y Aleatorio	Aleatorio	Aleatorio	1

Tabla 6. Configuración de las herramientas para la Falla de Tormenta Broadcast

HERRAMIENTA DE GENERACIÓN DE TRÁFICO	NIVEL DE VOLUMEN DE TRAFICO GENERADO [Utilización en KBytes]	PATRÓN DE TRÁFICO	TAMAÑO DE PAQUETE [Bytes]	No DE EQUIPOS GENERADORES
TF-GEN	0, 10, 50 Unicast	Continuo y aleatorio	Se auto-ajusta	1
	1000-50000 Broadcast			
TF-GEN	0, 10, 50 Unicast	Continuo y aleatorio	Se auto-ajusta	1
IP NETWORK BROWSER; MAC ADDRESS DISCOVERY; PING SWEEP; SNMP SWEEP.	-	Aleatorio	Aleatorio	2

4.2.3 Esquema físico. Para la implementación de los esquemas de prueba planteados, se cuenta con los siguientes equipos del Laboratorio de Redes de la Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones de la Universidad Industrial de Santander, ubicado en el edificio de Alta Tensión salón 201.

- 1 Switch AVAYA CAJUN P333R
- 1 Hub AVAYA ARGENT BRANCH
- 10 computadores OPTIPLEX GX 260 marca DELL[64]

El Switch utilizado en el segmento de prueba controla el flujo de tráfico entre los dispositivos conectados. Al conectar los equipos que se encargarán de la generación de tráfico y los equipos cliente en puertos separados del switch, se produce una carga extra para el dispositivo de red, pero esto no tiene ningún impacto para los clientes; lo cual no correspondería al escenario que se quiere plantear, en el cual las fallas representan una disminución en la calidad del servicio para los usuarios que hacen parte del segmento.

Una mejor opción consiste en usar un Hub para establecer el segmento de prueba. A este dispositivo se conectan múltiples estaciones, que se convertirían en los clientes y se completa el esquema mediante la conexión del Hub a un puerto del Switch.

Este puerto del Switch será el punto representativo del segmento donde se obtendrán los datos del comportamiento de las variables MIB mediante el proceso de sondeo.

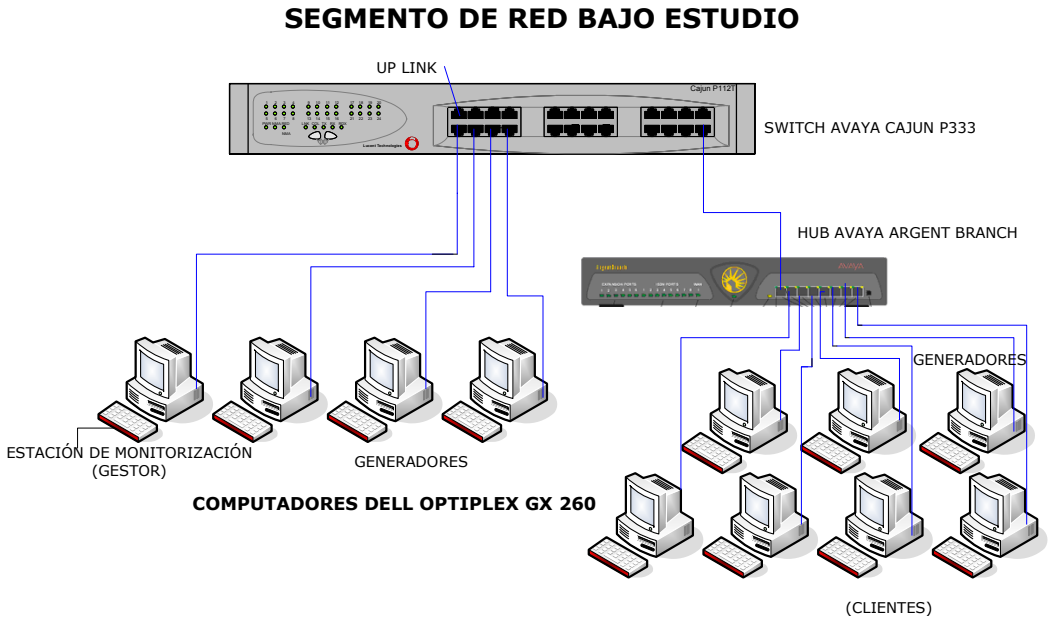
Para que este esquema de prueba permita obtener los mejores resultados, es necesario ubicar múltiples generadores de tráfico tanto en el mismo segmento en que están ubicados los clientes, como por fuera de él.

[64] Una descripción detallada de las características de estos equipos se encuentra en el anexo E.

Este tipo de configuración ayuda a generar cargas variables de tráfico en las diferentes partes de la red, que afectarán el desempeño en el segmento de prueba.

Teniendo en cuenta estas consideraciones se estableció el siguiente esquema para el segmento de prueba.

Figura 8. Esquema del segmento de prueba elegido.



Ya que la generación de tráfico se realiza con herramientas software, cualquiera de los equipos clientes puede configurarse como generador, lo que permite establecer flujos de tráfico desde múltiples fuentes simultáneamente con lo cual se facilita la emulación de las condiciones que originan los escenarios de falla.

4.2.4 Análisis de variables soportadas por los dispositivos. Con la ayuda de la herramienta MIB Walk de Solarwinds se puede determinar cuáles variables de la MIB son soportadas por el dispositivo administrable disponible en el laboratorio, y tener certeza de que pueden ser utilizadas para obtener información al realizar las pruebas; los resultados obtenidos con dicha herramienta muestran que existen algunas variables seleccionadas

previamente que no son soportadas por los equipos de red a utilizar; esto indica que se debe realizar un ajuste al grupo de variables seleccionadas inicialmente en este trabajo.

Teniendo en cuenta esta situación y cuyos resultados son mostrados en las tablas 7 y 8, donde son presentadas las variables seleccionadas a partir del grupo elegido inicialmente, que serán utilizadas para verificar la asociación planteada previamente para cada falla.

Tabla 7. Grupo final de variables de la MIB asociadas a la falla de Tormenta Broadcast

TORMENTA BROADCAST		
NOMBRE	**OID**	**DESCRIPCIÓN**
IfInNUcastPkts	1.3.6.1.2.1.2.2.1.12	Muestra el número de paquetes entregados por esta subcapa a una superior, los cuales fueron direccionados a una dirección Broadcast o multicast en esta subcapa.
IfOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	Muestra el número total de paquetes que los protocolos de capas altas solicitaron para ser transmitidos y que fueron direccionados a una dirección broadcast o multicast en esta subcapa; incluyendo aquellos que fueron descartados o no enviados.

Tabla 8. Grupo final de variables de la MIB asociadas a la falla de Congestión

CONGESTIÓN		
NOMBRE	**OID**	**DESCRIPCIÓN**
IfInUcastPkts	1.3.6.1.2.1.2.2.1.11	Muestra el número de paquetes entregados por esta subcapa a una superior, los cuales no fueron direccionados a una dirección Broadcast ni multicast en esta subcapa.
IfOutUcastPkts	1.3.6.1.2.1.2.2.1.17	Muestra el número total de paquetes que los protocolos de capas altas solicitaron para ser transmitidos y que no fueron direccionados a una dirección broadcast o multicast en esta subcapa; incluyendo aquellos que fueron descartados o no enviados.
EtherStatsOctets	1.3.6.1.2.1.16.1.1.1.4	Muestra el número total de octetos de datos (incluyendo aquellos en paquetes defectuosos) recibidos en la red (excluyendo los bits de entramado pero incluyendo octetos FCS). Esta variable puede ser usada como un estimado razonable de la utilización de Ethernet.

Entre el conjunto de variables planteados inicialmente existen tres que no aportan información específica sobre el comportamiento del tráfico en la red, pero son variables que indican la presencia de fallas, y son muy útiles para complementar la información suministrada por las variables que se seleccionaron.

De esta forma se tendrá en cuenta el comportamiento de estas variables como una referencia para las pruebas a realizarse. Estas variables auxiliares se describen en la siguiente tabla.

Tabla 9. Variables de la MIB auxiliares

NOMBRE	OID	DESCRIPCIÓN
IfInDiscards	1.3.6.1.2.1.2.2.1.13	Muestra el número de paquetes entrantes que fueron elegidos para ser descartados aunque no se hayan detectado errores, para evitar que sean entregados a un protocolo de capa superior. Una posible razón para descartar tales paquetes es la liberación de espacio en los buffer.
IfInErrors	1.3.6.1.2.1.2.2.1.14	Para interfaces orientadas a paquetes, muestra el número de paquetes de entrada que contienen errores, evitando su entrega a protocolos de capa superior. Para interfaces orientadas a caracteres o de longitud fija, muestra el número de unidades de transmisión de entrada que contienen errores, evitando su entrega a protocolos de capa superior.
EtherStatsCollisions	1.3.6.1.2.1.16.1.1.1.13	Indica el número total de colisiones mejor estimado (más preciso) en este segmento Ethernet. El valor retornado dependerá de la localización del sondeo RMON.

5. ANÁLISIS Y RESULTADOS DE LAS PRUEBAS REALIZADAS

En este capítulo se presentan los resultados obtenidos mediante la monitorización de las variables MIB seleccionadas siguiendo los esquemas descritos en el capítulo 4. Se realizó un análisis que tuvo como fin verificar que las relaciones establecidas entre las variables de la MIB y las condiciones de falla son válidas.

5.1 ANTECEDENTES

La constante evolución e incremento en el uso de las redes de computadores en todos los ámbitos, ha hecho necesario que los sistemas de gestión estén en capacidad de detectar y diagnosticar potenciales problemas y fallas dentro de ellas, para mantener la disponibilidad, la confiabilidad, y asegurar un nivel de calidad de servicio adecuado.

Estas fallas y anomalías han sido detectadas utilizando diversas técnicas como la inteligencia artificial, los sistemas expertos, técnicas avanzadas de bases de datos, métodos probabilísticos y el modelamiento mediante maquinas de estado finito[65].

5.1.1 Algoritmos de detección. Los procedimientos de detección de fallas hacen parte de las herramientas con las que cuenta un administrador o gestor de red para asegurar la confiabilidad y disponibilidad de ella. Estos procedimientos surgen comúnmente por la combinación del conocimiento previo sobre la configuración de la red y de la información obtenida sobre las características del tráfico existente. Su objetivo es la detección e identificación de las anomalías y cuellos de botella presentes en la red.

La importancia del proceso de detección dentro de cualquier esquema de gestión de redes, radica en que una vez realizada la detección de estas anomalías, pueden ser establecidas

[65] THOTTAN, M; JI, Chuanyi. "Anomaly Detection in IP Networks", IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL 51, NO. 8, AUGUST 2003.

alarmas dentro del sistema de gestión de red utilizado, lo cual facilita el rápido despliegue de herramientas de recuperación ante la presencia de estas fallas.

Un sistema de detección de fallas básicamente tiene implementados algoritmos de detección que identifican variaciones abruptas en el comportamiento estadístico normal de la red. La dificultad principal para la implementación de estos algoritmos y en general para el funcionamiento de los sistemas de detección de fallas, es que no existen modelos estadísticos precisos sobre la operación normal de las redes basadas en IP, ya que debido a su alta variabilidad se dificulta el entendimiento pleno de la dinámica que las rige, lo que determina que el modelamiento preciso del comportamiento normal de las redes sea todavía un campo activo de investigación.

Las aproximaciones o métodos más comunes para la implementación de estos algoritmos de detección son las aproximaciones basadas en reglas, los modelos de máquina de estado finito, el ajuste de patrones y el análisis estadístico.

- **Aproximaciones basadas en reglas.** En los inicios de la investigación en este tema, los esfuerzos se orientaban a la creación de sistemas expertos[66] para la detección de anomalías o fallas de red. Esta clase de sistemas estaban formados por una voluminosa base de datos, en la cual estaban consignadas las reglas de comportamiento de una red en estado de falla. Esta base de datos luego era utilizada para determinar la ocurrencia de una falla.

Otra forma de aproximación basada en reglas es la creación de algoritmos de detección mediante el establecimiento de umbrales, donde el factor clave no es el valor de los umbrales superiores e inferiores, sino la información de cuando una variable de estudio excede estos límites[67]. La principal limitante de este tipo de aplicaciones es que los

[66] Bajo el término de Sistemas Expertos se entiende un nuevo tipo de software que imita el comportamiento de un experto humano en la solución de un problema. Pueden almacenar conocimientos de expertos para un campo determinado y solucionar un problema mediante deducción lógica de conclusiones.

[67] HOOD, Cynthia S; JI, Chuanyi. Intelligent Agents For Proactive Fault Detection, IEEE Internet Computing, March - April 1998.

umbrales son dependientes del nivel de tráfico en la red, lo que constituye un problema en el momento de fijar el nivel de estos. La importancia de este aspecto radica en que la precisión y sensibilidad del proceso de detección depende del nivel fijado para los umbrales.

En general las aplicaciones basadas en reglas resultan muy lentas para su implementación en tiempo real y dependen excesivamente del conocimiento previo de las condiciones de falla en la red bajo estudio, limitando el rango de acción de los algoritmos que siguen esta orientación a la experiencia del administrador de red e impidiendo su adaptación a los cambios que se presenten.

- **Modelos de máquinas de estado finito.** Las máquinas de estado finito[68] son diseñadas para una falla específica, de la cual se tiene un completo conocimiento previo mediante los datos históricos. Los objetivos perseguidos con su implementación son muy amplios, ya que no solo se persigue la detección del comportamiento anómalo, sino la identificación y diagnóstico del problema que origina la falla.

Los estados de la máquina de estado finito son modelados a partir de las secuencias de alarmas obtenidas desde diferentes puntos de la red. Estas alarmas deben contener información sobre el nombre del dispositivo sondeado, los síntomas detectados y el tiempo en el que ocurrió el evento.

La mayor dificultad para la implementación de algoritmos de detección basados en este tipo de aplicación, es que no todas las fallas pueden ser tomadas a partir de una secuencia finita de alarmas de longitud razonable. Esta situación obliga a que el número de parámetros a ser aprendidos se incremente exageradamente y a su vez queden sin validez en el momento en que se presente cualquier cambio en la configuración de la red. De esta forma cualquier tipo de variabilidad presente en la red exigirá la obtención de un gran conocimiento fuera de

[68]Una máquina de estado finito es una herramienta abstracta que se utiliza para reconocer un determinado lenguaje regular. Es un modelo matemático de un sistema que recibe una cadena constituida por caracteres de cierto alfabeto y determina si esa cadena pertenece al lenguaje que el autómata reconoce.

línea (off-line) antes de estar en capacidad de poner en funcionamiento el sistema de detección en la red.

- **Ajuste de patrones[69] (Pattern Matching).** Esta técnica tiene como principio la descripción de las anomalías o fallas como desviaciones del comportamiento estadístico normal. El objetivo principal de este tipo de aplicaciones es enfrentar la alta variabilidad de los entornos de red.

A diferencia de los esquemas presentados anteriormente, en el ajuste de patrones se busca la construcción de los perfiles de tráfico de las redes a partir del entrenamiento en línea del sistema de detección. Esta tarea es puesta en práctica mediante la utilización de vectores de características asociados a síntomas de falla específicos como la utilización de enlace, la pérdida de paquetes y el número de colisiones.

Los perfiles de tráfico obtenidos, son luego caracterizados teniendo en cuenta el período de tiempo específico (hora del día, día de la semana, etc.), estableciendo así una línea de comportamiento base que será comparada con los nuevos datos adquiridos; gracias a este proceso de comparación, se identifican las anomalías mediante la detección de los puntos de datos que no se ajusten a los rangos fijados en la línea base.

Estos rangos o límites de tolerancia son fijados fundamentándose en diferentes niveles de desviación estándar de los datos, y deben ser comprobados mediante un análisis de datos riguroso.

En los algoritmos de detección desarrollados siguiendo esta orientación, su eficiencia depende de la precisión con la que se construya el perfil de tráfico. La principal desventaja de este hecho es la gran cantidad de tiempo que debe ser dedicada a la generación de estos perfiles de tráfico, especialmente en una red totalmente nueva.

[69] Clase de algoritmos en los que se intenta hallar el patrón que coincide con una muestra de referencia (o el que más se asemeja a la misma) a través de un proceso de emparejamiento o alineación

- **Análisis estadístico.** Al crear algoritmos de detección basados en el conocimiento previo o el modelamiento del comportamiento normal de las redes, se afronta la necesidad de recalibrar o reentrenar el sistema de detección ante cualquier cambio de configuración en la red.

Ante esta problemática, la combinación de entrenamiento en línea y análisis estadístico se convierte en una opción efectiva, para mantener un seguimiento continuo del comportamiento de la red.

Los algoritmos de detección basados en el análisis estadístico, han sido usados frecuentemente tanto para el rastreo de fallas de red como de violaciones a la seguridad de estas.

Para la implementación de esta clase de algoritmos es necesaria la consecución de una gran cantidad de información acerca de los escenarios o condiciones específicas de cada falla que se quiera detectar, ya que cada uno de estos escenarios difiere en sus características y manifestaciones dentro de las operaciones comunes de una red. La principal fuente para este tipo de información son las operaciones de gestión realizadas mediante el protocolo SNMP, cuyos resultados son almacenados en las diferentes bases de información de gestión (MIB).

Se puede apreciar que para la implementación de sistemas de gestión prácticos, es vital la creación de algoritmos o herramientas de detección de fallas, en las cuales se complemente el conocimiento de la red bajo estudio con las propiedades estadísticas de la información ofrecida por la propia red.

Para la implementación de estos algoritmos de detección en tiempo real, se debe recurrir a la inteligencia artificial, ya que la ausencia de modelos precisos del comportamiento normal de las redes obliga a evaluar las condiciones de la red en el tiempo y exige niveles de cómputo de alta complejidad.

5.2 IMPLEMENTACIÓN DE ALGORITMOS DE DETECCIÓN

Los algoritmos de detección, además de hacer parte de las herramientas con las que cuentan los sistemas de gestión prácticos, pueden ser utilizados en la investigación de la operación de la red, ya que poseen la capacidad de rastrear características específicas del comportamiento y propiedades de las redes. Esto ofrece grandes ventajas a los investigadores, que pueden concentrarse sólo en los aspectos de mayor interés para sus objetivos.

El objetivo principal de este trabajo era el desarrollo de una metodología para la verificación de las relaciones planteadas entre las condiciones de las fallas de red bajo estudio y las variables MIB seleccionadas previamente; debido a esto se decidió desarrollar la implementación de algoritmos de detección basados en las propiedades estadísticas de los datos obtenidos a partir de las variables MIB elegidas, previamente tabulados en hojas de cálculo, como el medio para verificar las relaciones planteadas.

Los datos obtenidos a partir de los esquemas de prueba planteados en el capítulo anterior, son evaluados mediante los algoritmos de detección desarrollados, permitiendo así la realización de un proceso de análisis, que conducirá a la verificación y validación de las relaciones Condiciones de falla-Variables MIB trazadas al inicio de este trabajo.

5.3 DESCRIPCIÓN DE LOS ALGORITMOS DE DETECCIÓN IMPLEMENTADOS

Después de analizar los antecedentes en esta área, se determinó implementar tres tipos de algoritmos, que faciliten la detección de alteraciones abruptas[70] en los valores de las variables MIB bajo estudio.

[70] THOTTAN, M; JI, Chuanyi. "Anomaly Detection in IP Networks", IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL 51, NO. 8, AUGUST 2003.

A cada falla le ha sido asociado un grupo de variables MIB. Cada variable dentro de este grupo se relaciona directamente con la definición de la falla y por lo tanto para el resultado positivo en la detección de la falla se debe tener en cuenta un análisis de detección individual para cada variable asociada; cualquier detección positiva[71] en una variable dentro del grupo asociado es indicativo de anomalías relacionadas con la falla. El algoritmo implementado debe tener en cuenta este análisis individual de variables MIB detectando cambios abruptos, y complementarse con los resultados de detecciones individuales en las demás variables elegidas dentro del grupo que puedan indicar una condición de falla.

En la descripción de los algoritmos de detección, los 3 tipos de algoritmos implementados tienen en común las siguientes características:

- El proceso de detección de anomalías se realiza simultáneamente sobre cada variable MIB dentro el grupo asociado a la falla. Los resultados de detección individuales son llevados a una compuerta OR la cual indica de una forma general la detección de las alarmas.

- El proceso de detección debe ser causal[72] para que pueda ser observado como un procedimiento en tiempo real.

- Para realizar la detección de una anomalía se establecen envolventes, una para un nivel superior y otra para un nivel inferior. Estas envolventes definen franjas con límites que establecen la generación de alarmas a nivel de variable para cada muestra que los sobrepasa. Estas muestras son consideradas puntos de anomalía y cada una de las alarmas asociadas a estos es establecida mediante la comparación entre el valor del dato actual y el valor de la envolvente.

[71] Una detección positiva ocurre cuando el algoritmo implementado detecta un cambio significativo en los valores de una variable.

[72] OPPENHEIM, Alan V; WILLSKY, Alan S. SEÑALES Y SISTEMAS. 2da ed, Prentice Hall Hispanoamericana, 1998. Pág. 46.

Para establecer los niveles de las envolventes, los datos originales deben ser depurados inicialmente mediante un proceso de filtrado.

Descripción del filtro. Su finalidad es suavizar la curva de los datos originales para que al establecer las envolventes, estas no muestren el comportamiento de alta variabilidad del segmento de red. El diseño de este filtro se basa en el valor de la mediana calculada para una ventana móvil que se desplaza sobre los datos de entrada; cada vez que la ventana móvil realiza un desplazamiento un nuevo dato ingresa en el filtro como primer elemento dentro de la ventana, a medida que la ventana se desplaza se va actualizando el valor de la mediana creando una curva suavizada a la salida del filtro.

La ventana móvil mencionada se define como un vector obtenido a partir de los datos de entrada:

Si $X[n] = x_1, x_2, x_3, \ldots, x_i, \ldots, x_n$ es el vector de datos de entrada y x_i el valor del dato actual, entonces una ventana móvil de tamaño k muestras será definida por el vector $V[k] = x_i, x_{i-1}, x_{i-2}, x_{i-3}, \ldots, x_{i-k-1}$.

La mediana[73] del vector $V[k]$ es una medida de tendencia central para los datos dentro de la ventana, si el valor de k es impar, la mediana es un dato que pertenece a la muestra el cual divide los datos de la ventana en dos partes iguales; si k es par, la mediana se obtiene a partir del valor medio de los dos datos centrales.

Con los valores de mediana obtenidos para cada desplazamiento de la ventana, se crea un vector $Y[m]$ de tamaño[74] $m = n-(k-1)$ muestras con las características de la curva que

[73] MONTGOMERY, Douglas C; RUNGER, George C. PROBABILIDAD Y ESTADÍSTICA APLICADAS A LA INGENIERÍA, Segunda Edición, Editorial Limusa S.A. 2002. Pág. 30

[74] El tamaño de este vector es determinado por un offset de $k-1$ datos en el vector de entrada; en este rango de muestras no se podrán establecer detecciones debido al punto de inicio de datos (muestra k) desde el cual son tomadas en cuenta las muestras que conforman la ventana; para solucionar este problema, se dejó un margen amplio de muestras de monitorización antes de que se iniciaran las pruebas de generación de las fallas.

describe a los datos originales, pero eliminando picos y suavizándola.

- **Envolventes.** La curva de la envolvente se debe adaptar a las condiciones de variación típicas del segmento de red para tratar que la mayoría de datos que no representen cambios abruptos queden por dentro de ella y que los datos que estén muy alejados y representen anomalías sean detectados; para esto se utilizó la desviación estándar como medida de dispersión para describir la variabilidad de los datos.

La desviación estándar[75] σ esta definida para una muestra de ***n*** observaciones, $x_1, x_2, x_3, \ldots, x_i, \ldots, x_n$ por la ecuación (1).

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}\left(x_i - \overline{x}\right)^2}{n-1}} \quad (1)$$

El ancho de la envolvente es ajustado al establecer un factor "**a**" de desviaciones estándar a partir del resultado a la salida del filtro (vector ***Y[m]***) de esta forma la envolvente superior se obtiene a partir de la ecuación (2).

$$Y[m] + \sigma \cdot a \qquad (2)$$

Y la envolvente inferior a partir de la ecuación (3).

$$Y[m] - \sigma \cdot a \qquad (3)$$

Generalmente se utiliza en las áreas de estadística y control de calidad estadístico[76] un

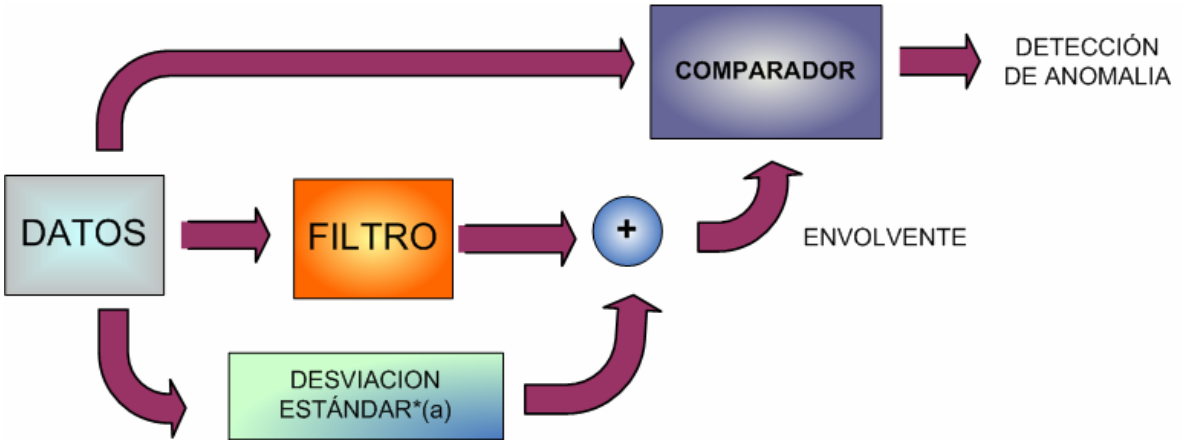
[75] MONTGOMERY, Douglas C; RUNGER, George C. PROBABILIDAD Y ESTADÍSTICA APLICADAS A LA INGENIERÍA, Segunda Edición, Editorial Limusa S.A. 2002. Pág. 5

[76] FEATHER, Frank; SIEWIOREK, Dan; MAXION, Roy. "Fault Detection in an Ethernet Network Using Anomaly Signature Matching", SIGCOMM 1993. Pág. 3

factor “a” de tres desviaciones estándar para fijar una envolvente superior cuando la distribución es normal.

El diagrama de bloques general del algoritmo que describe las diferentes etapas del proceso de detección es presentado en la figura 9.

Figura 9. Esquema básico del algoritmo de Detección.



5.3.1 Propiedades específicas de los detectores implementados. En esta sección serán presentadas las diferencias existentes entre los tres tipos de detectores implementados en este estudio, ya que la forma particular de agrupar los datos para calcular la desviación estándar por parte de cada uno de ellos altera sensiblemente el establecimiento de las envolventes.

Cada uno de estos detectores debe ser ajustado mediante la variación del tamaño ***k*** de la ventana móvil y el factor ***a*** de la desviación estándar para fijar el ancho de las envolventes, en este punto el objetivo es no dejar pasar ningún cambio abrupto que pueda indicar una condición de falla; por tal razón el sistema debe ser bastante sensible, sin importar que el número de falsas detecciones sea elevado interfiriendo con el desempeño del algoritmo.

La herramienta utilizada para poner en práctica los algoritmos fue LabView 6.1 debido a la

facilidad de programación y manipulación de parámetros de entrada, además se acopla sencillamente con otros programas de soporte utilizados como MATLAB 6.5 y EXCEL.

Después de realizar varias correcciones para tratar de ajustar los algoritmos mediante prueba y error; los tres detectores se implementaron de la siguiente manera:

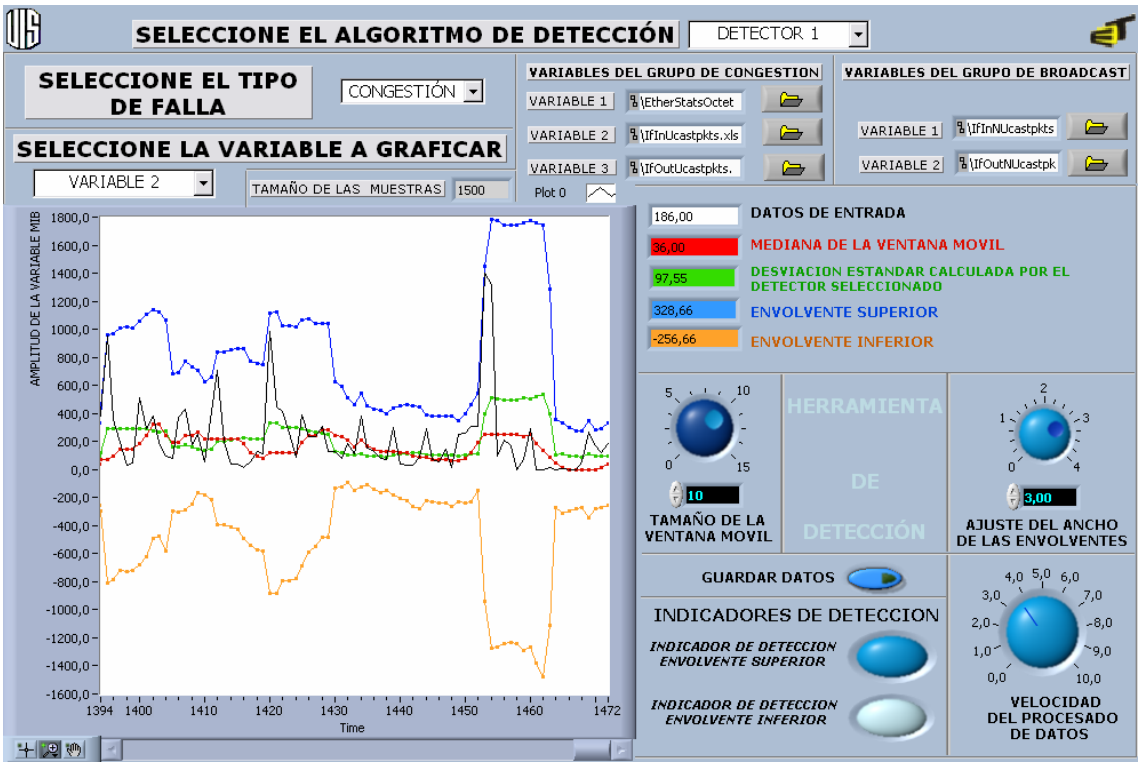
- **Detector Nº 1.** Realiza el cálculo de la desviación estándar usando la misma ventana móvil que utiliza el filtro.

A partir de una serie de pruebas en las variaciones de los parámetros (***k*** y ***a***) de configuración para este detector, se decidió utilizar un factor ***a*** = 3 desviaciones estándar para establecer el ancho de las envolventes, y para el tamaño de la ventana móvil del filtro se definió un ***k*** = 10 muestras que representan 20 minutos de tiempo; el valor de ***k*** se ajusta de manera que no sea tan pequeño que el filtro actué como un pasa-todo o tan grande que se pierda información debido al retardo que se generaría.

En la figura 10 se observa la interfaz gráfica que se utilizó para el desarrollo de los algoritmos de detección y en la cual se presenta un análisis para el detector N°1. Esta Interfaz ofrece la opción de seleccionar el tipo de algoritmo de detección, el tipo de falla que se quiera analizar y la variable a mostrar en la ventana grafica, en las perillas ubicadas a la derecha se ajusta el valor del tamaño de la ventana móvil y del factor "***a***" de desviaciones estándar. En las curvas mostradas se pueden apreciar los datos de entrada para la variable 2 (IfInUcastPkts) (negro), la desviación estándar calculada por el detector seleccionado (verde), su mediana de la ventana móvil (rojo), y las envolventes (superior- azul, inferior- naranja).

Los datos tabulados por variable se importan desde los archivos de hoja de cálculo (EXCEL), obtenidos del software de monitorización mediante los iconos que se aprecian en la parte superior derecha de la interfaz gráfica. También se tiene la opción de guardar los resultados del proceso de detección del grupo de variables directamente en hojas de cálculo.

Figura 10. Interfaz gráfica del Detector Nº 1.



Datos de entrada

Mediana de la ventana móvil

Desviación estándar calculada por el detector N°1

Envolvente Superior

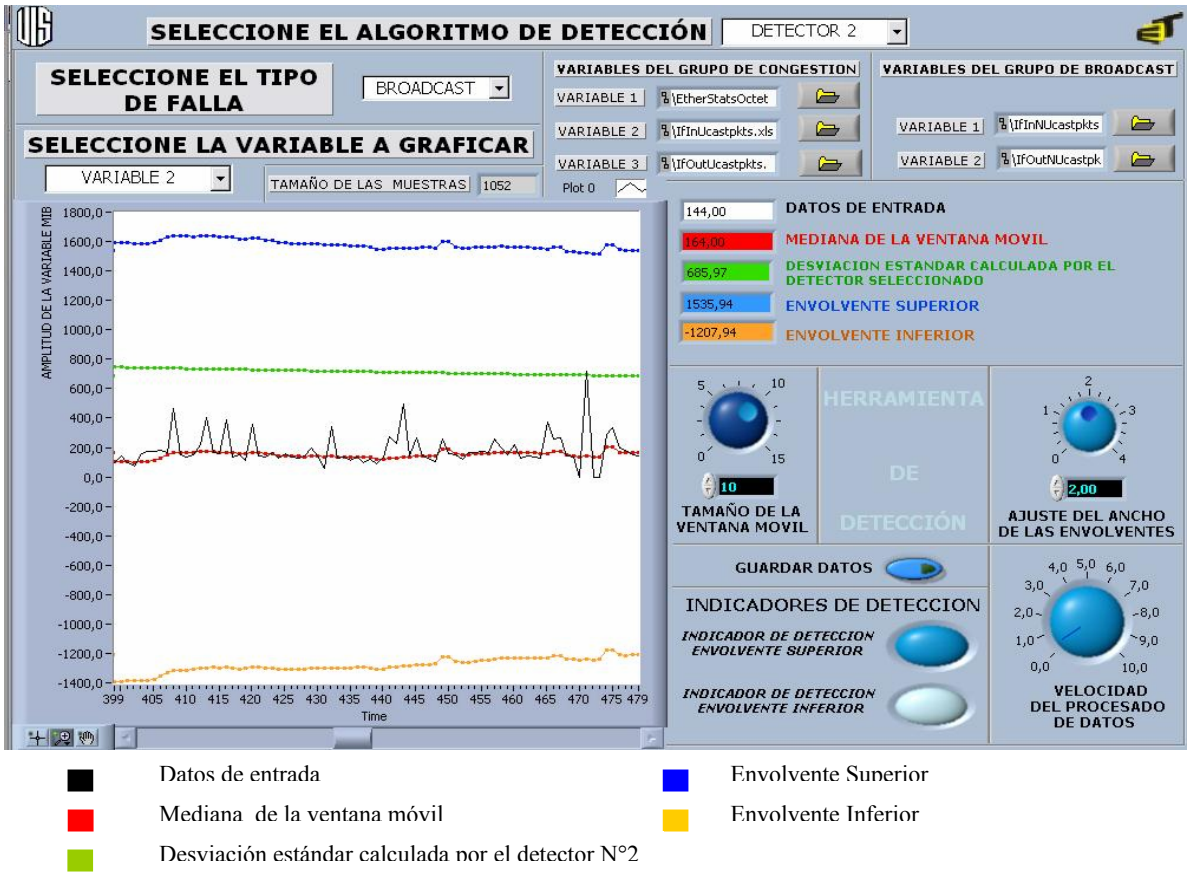
Envolvente Inferior

▪ **Detector Nº 2.** Realiza el cálculo de la desviación estándar para el registro histórico de los datos; es decir que la medida de dispersión para un nuevo dato es tomada a partir de todos los datos anteriores a este. Las envolventes en este detector se acomodan a la variabilidad que se va presentando en los datos, por lo cual se pueden presentar falsas alarmas en el inicio del análisis de los datos; la desviación estándar y las envolventes tienden a mantenerse constantes a lo largo del procesado de los datos.

Por las mismas razones que el detector anterior, para el filtro se definió como tamaño para la ventana móvil un $k = 10$ muestras y para establecer el ancho de las envolventes se utilizó un factor $a = 2$ desviaciones estándar.

En la figura 11 se observa la interfaz gráfica implementada para el análisis realizado a la falla de tormenta de broadcast utilizando el detector N°2; las características de la interfaz gráfica en este caso son iguales a las del detector Nº 1.

Figura 11. Interfaz gráfica del Detector Nº 2.

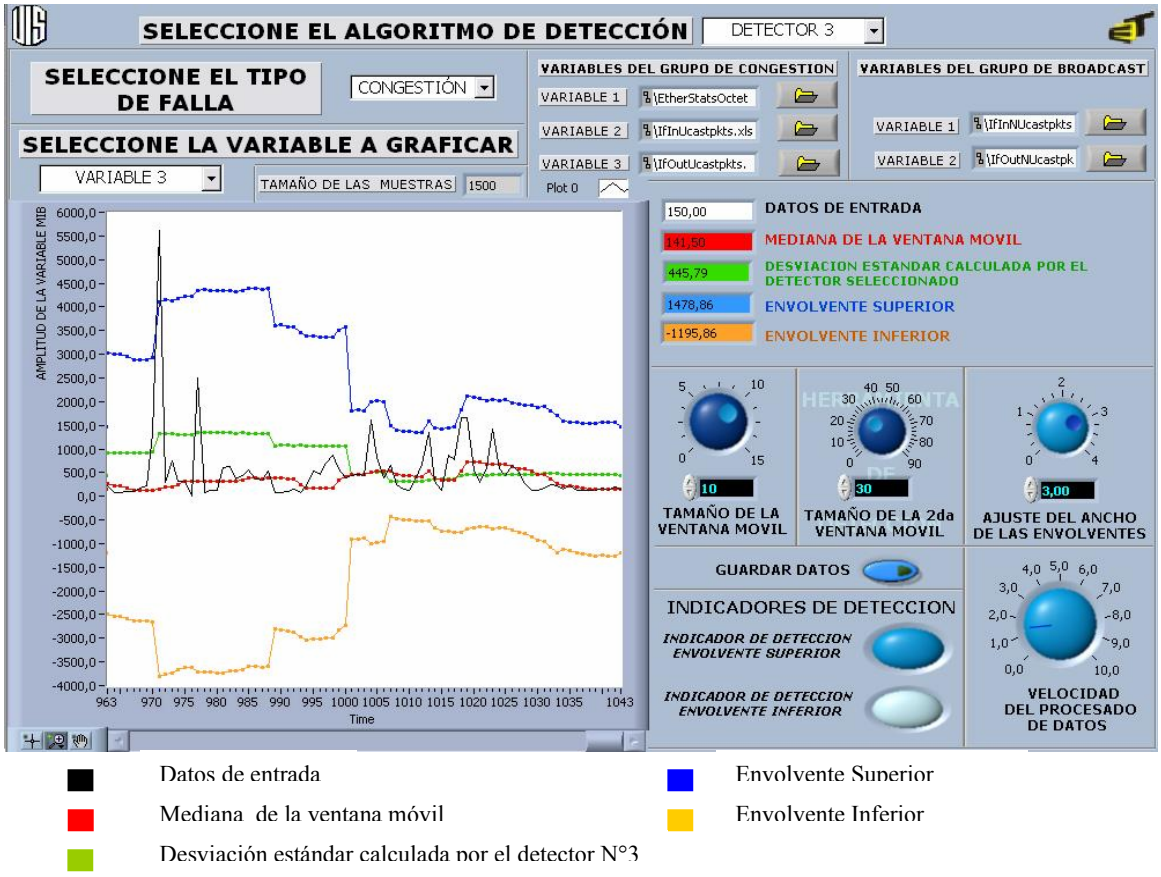


Datos de entrada
Mediana de la ventana móvil
Desviación estándar calculada por el detector N°2
Envolvente Superior
Envolvente Inferior

- **Detector Nº 3.** Realiza el cálculo de la desviación estándar a los datos contenidos dentro de una segunda ventana móvil; este detector es una combinación de los dos anteriores, ya que la idea es establecer para el cálculo de la desviación estándar una base de datos con un tamaño mayor k_2 el cual se fijo en 30 muestras o una hora de monitorización; de la misma forma que los detectores anteriores, el tamaño para la ventana del filtro en este caso k_1 se definió en 10 muestras y para establecer el ancho de las envolventes se utilizó un factor a = 3 desviaciones estándar.

En la figura 12 se observa la interfaz gráfica implementada para el análisis realizado a la falla de congestión utilizando el detector N°3 y su descripción es igual que en el detector Nº 1; en este detector aparece una nueva perilla donde se puede ajustar el tamaño para la segunda ventana móvil.

Figura 12. Interfaz gráfica del Detector Nº 3.



Después de que una falla ocurre es muy difícil estimar el comportamiento de las variables inmediatamente se presenta la anomalía; en la mayoría de trabajos presentes en la literatura, el estudio de la duración de la falla, ubicación de la falla etc., se realiza en un nivel más alto que el nivel de variable en el cual se basa este estudio. En niveles superiores se utilizan parámetros complementarios entre los que destacamos los siguientes:

- Se va más allá de la información obtenida por variable y se establecen relaciones entre ellas[77] como rangos de valores de sus magnitudes.

- Se implementan agentes distribuidos en diferentes nodos de la red.

- Dependiendo de la severidad de la falla se establecen traps[78] o mensajes *syslog*[79] con el fin de estimar la duración de la falla y rastrear sus efectos.

Debido a la incapacidad de modelar el comportamiento de las variables ya que son de característica aleatoria, no se puede garantizar que el comportamiento estadístico de estas, siga un patrón que pueda ser usado en la implementación del algoritmo de detección para determinar la duración de las fallas.

En este punto es donde entran en juego las variables auxiliares[80] definidas en el capítulo anterior, las cuales no son procesadas con los detectores pero ayudan al administrador a vigilar el buen funcionamiento de la red.

Generalmente la envolvente inferior se diseña con el fin de detectar caídas en los valores de las variables MIB. Cuando el contador correspondiente a la variable no presenta cambios, se asocia a fallas de estado del dispositivo o a problemas físicos los cuales no se han tenido en cuenta en este estudio por su baja incidencia en una red bien diseñada y por lo tanto no se presentarán resultados para detección de caídas en los valores de las variables asociadas a las fallas.

---

[77] THOTTAN, M; JI, Chuanyi. "Proactive Anomaly Detection Using Distributed Intelligent Agents", IEEE Network September/ October 0890-8044/ 1998. Pág. 4

[78] Mensajes de notificación de anomalías no solicitados.

[79] Una de las posibilidades que existen para detectar problemas en un sistema operativo, sea este servidor o no, es a través de las trazas o logs que generan las distintas aplicaciones que en él se ejecutan, incluyendo el propio kernel. Una traza no es más que un mensaje breve que avisa que un servicio inició; mientras que otras, en cambio, pueden indicar una emergencia grave, como puede ser un fallo físico en algún dispositivo; normalmente va acompañado de la fecha y hora en que se produce, el nombre de la máquina donde se produce y el programa que la origina. El Programa que implementa el servicio de trazas en ambientes UNIX se llama Syslog.

[80] Ver Tabla 9 Variables de la MIB auxiliares.

5.3.2 Desempeño de los algoritmos. El desempeño de los algoritmos se puede evaluar[81,82,83] en términos de la sensibilidad para caracterizar la tasa de falsas alarmas; para esto se han tenido en cuenta parámetros tales como la tasa de alarmas por hora, la probabilidad de detección P_D y la probabilidad de falsas alarmas P_F; los valores para P_D y P_F son estimados usando las ecuaciones (4) y (5) para cada tipo de falla.

$$P_D = \frac{Número\ total\ de\ comparaciones\ correctas}{Número\ total\ de\ eventos\ de\ inyección\ de\ tráfico} \qquad (4)$$

$$P_F = \frac{Número\ total\ de\ falsas\ alarmas}{Número\ total\ de\ muestras} \qquad (5)$$

El número total de comparaciones correctas se refiere al número de detecciones positivas de la herramienta que corresponden a eventos de inyección de tráfico.

El número total de falsas alarmas se obtiene de la resta entre el número total de alarmas y el número total de comparaciones correctas.

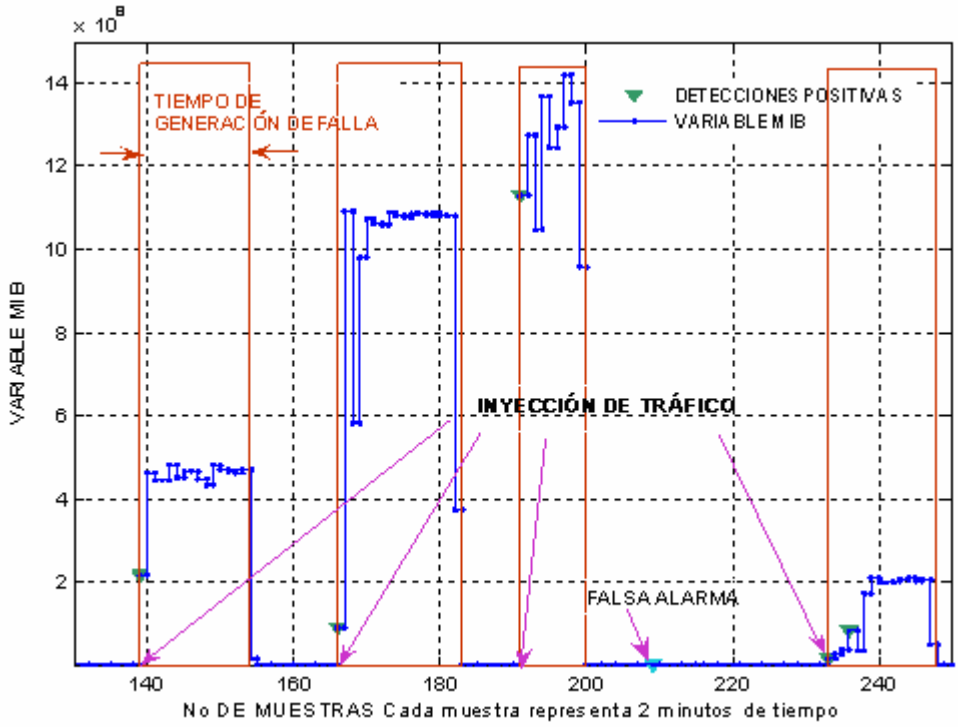
En la figura 13 se aprecian los parámetros usados en la etapa de evaluación del desempeño de los algoritmos de detección: el tiempo de generación de falla, la inyección de tráfico generado, y las falsas alarmas.

[81] FEATHER, Frank; SIEWIOREK, Dan; MAXION, Roy. "Fault Detection in an Ethernet Network Using Anomaly Signature Matching", SIGCOMM 1993. Pág. 9

[82] THOTTAN, M; JI, Chuanyi. "Adaptive Thresholding for Proactive Network Problem Detection", Rensselaer Polytechnic Institute 1999. Pág. 6

[83] HOOD, Cynthia S; JI, Chuanyi. Intelligent Agents For Proactive Fault Detection, IEEE Internet Computing, March - April 1998. Pág. 7

Figura 13. Escenario de detección de fallas.



La probabilidad de detección es una medida de las detecciones positivas sobre el total de fallas presentadas, lo cual indica el grado de sensibilidad del algoritmo para rastrear las anomalías.

5.4 RESULTADOS DE LA IMPLEMENTACIÓN DE LOS ALGORITMOS DE DETECCIÓN

Para la evaluación de los tres detectores implementados, se utilizaron los datos tabulados obtenidos de la monitorización durante las pruebas realizadas según los esquemas definidos anteriormente para una frecuencia de monitorización de 2 minutos, con esta frecuencia de monitorización se adquiere un mayor número de puntos de datos, se mejora el análisis estadístico y se obtiene mayor información sobre el comportamiento de la variable MIB[84], cada muestra contiene el incremento de los contadores de la variable MIB para la

[84] THOTTAN, M; JI, Chuanyi. "Anomaly Detection in IP Networks", IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL 51, NO. 8, AUGUST 2003. Pág. 12

frecuencia de monitorización seleccionada.

En algunos casos el nivel de tráfico inyectado hace que una detección positiva se mantenga por dos o más muestras, para el análisis del desempeño de los algoritmos sólo se tienen en cuenta la primera de ellas, ya que todas estas detecciones hacen parte de la misma prueba en la que se genera tráfico.

Como se presentó en el capítulo 4, se implementaron las pruebas establecidas en las tablas 5 y 6, combinando la dirección en la que el flujo de tráfico es generado desde los equipos clientes.

Los resultados obtenidos se muestran específicamente para el grupo de variables asociado a cada falla.

5.4.1 Resultados para la falla de congestión.

- **Características de los datos:**

Nº total de muestras: 1500 muestras
Frecuencia de monitorización: 2 minutos
Tiempo total de sondeo: 50 horas
Número fallas de congestión generadas: 17

Con los resultados de las detecciones para las tres variables seleccionadas para esta falla se obtuvieron los resultados de la tabla 10, el comportamiento de este grupo de variables se observa en las figuras 13 y 14, en el eje izquierdo se encuentran las variables que cuentan paquetes tanto de entrada y de salida (IfInUcastPkts, IfOutUcastPkts), en el eje derecho la variable EtherStatsOctets que cuenta octetos.

Tabla 10. Resultados de detección en los tres algoritmos implementados para el grupo de variables de la falla de congestión.

FALLA DE CONGESTIÓN					
	PORCENTAJE DE DETECCIONES DEL TOTAL DE MUESTRAS (Incluyendo falsas alarmas)	PROMEDIO DE DETECCIONES POR HORA	NÚMERO DE ANOMALIAS DETECTADAS POR LA ENVOLVENTE	P_D*100	P_F*100
DETECTOR No1	4.53%	1.36	17 / 17	100%	3.42%
DETECTOR No2	1.46%	0.44	15 / 17	88.2%	0.469%
DETECTOR No3	4.8%	1.44	16 / 17	94.1%	3.75%

Figura 14. Comportamiento del grupo de variables para la falla de congestión.

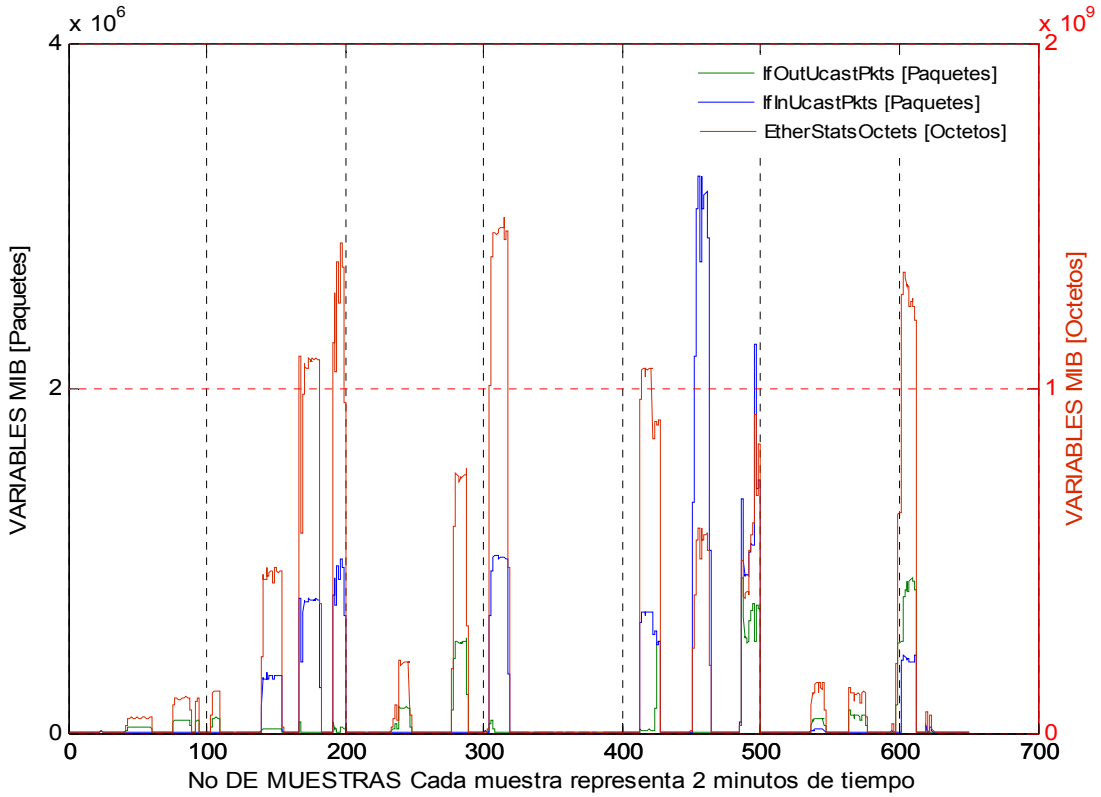
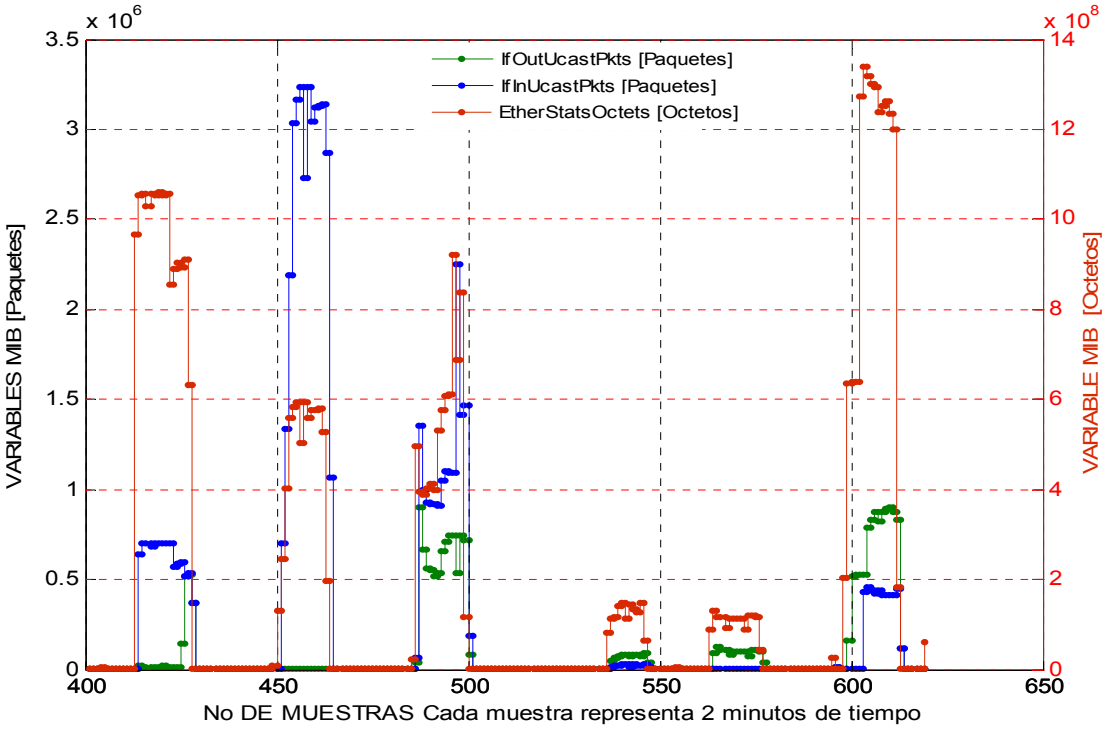


Figura 15. Comportamiento del grupo de variables para la falla de congestión para 7.3 horas de monitorización.



Como se esperaba debido a la definición estándar de cada variable la cuenta de octetos como valor estadístico de tráfico de la interfaz es capaz de detectar todas las fallas de congestión generadas por medio del detector Nº 1 como se aprecia en la tabla 11 y en la figura 15; así mismo reuniendo las detecciones obtenidas por las dos variables que cuentan paquetes, estas son capaces de detectar todas las fallas generadas como se aprecia en la tabla 12 y en la figura 16.

Tabla 11. Resultados de detección en los tres algoritmos implementados para la variable EtherStatsOctets.

EtherStatsOctets para la falla de CONGESTIÓN					
	PORCENTAJE DE DETECCIONES DEL TOTAL DE MUESTRAS (Incluyendo falsas alarmas)	PROMEDIO DE DETECCIONES POR HORA	NÚMERO DE ANOMALÍAS DETECTADAS POR LA ENVOLVENTE	P_D*100	P_F*100
DETECTOR No1	3.26%	0.98	17/17	100%	2.13%
DETECTOR No2	0.86%	0.26	12/17	70.5%	0.66%
DETECTOR No3	2.86%	0.86	12/17	70.5%	2.066%

Figura 16. Comportamiento de la variable EtherStatsOctets para la falla de congestión.

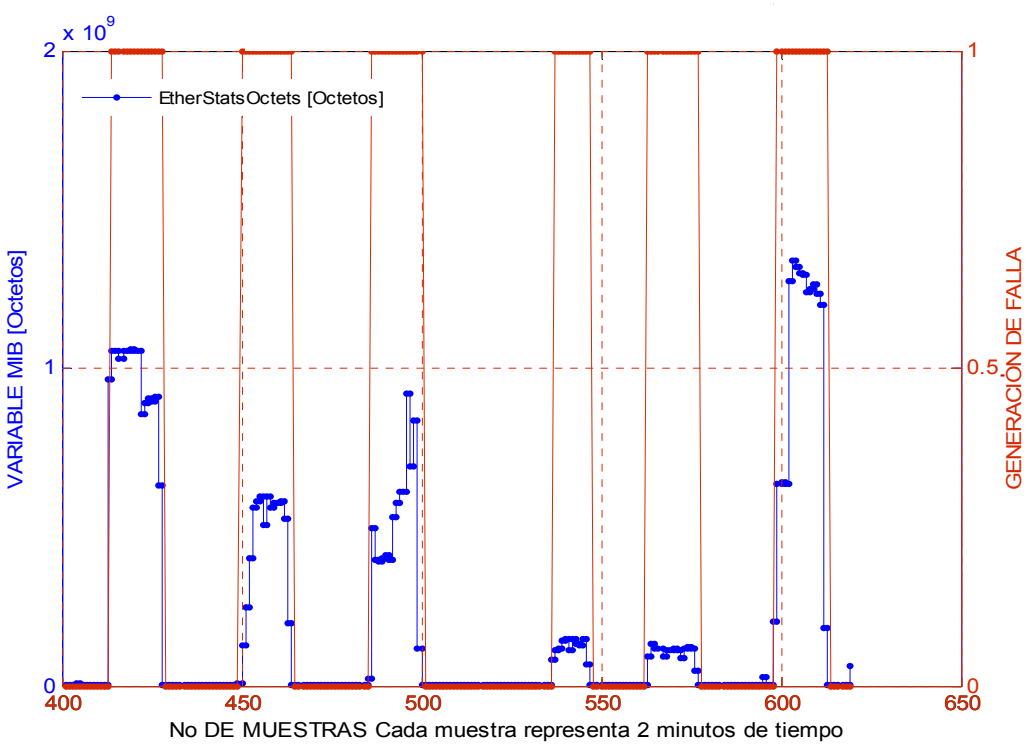
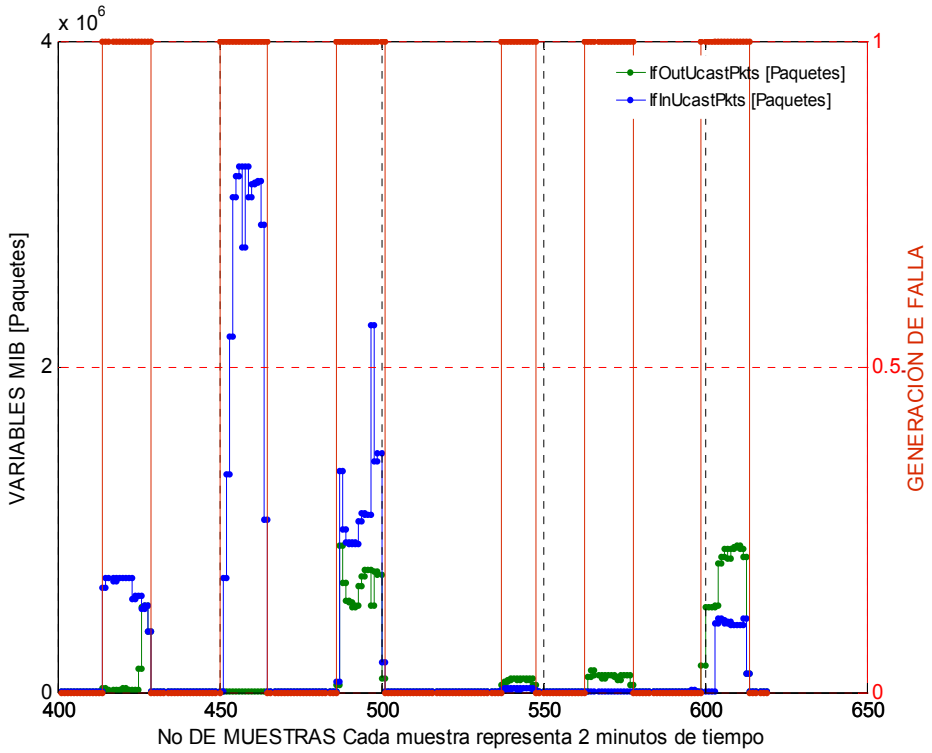


Tabla 12. Resultados de detección en los tres algoritmos implementados para la variable IfInUcastPkts e IfOutUcastPkts.

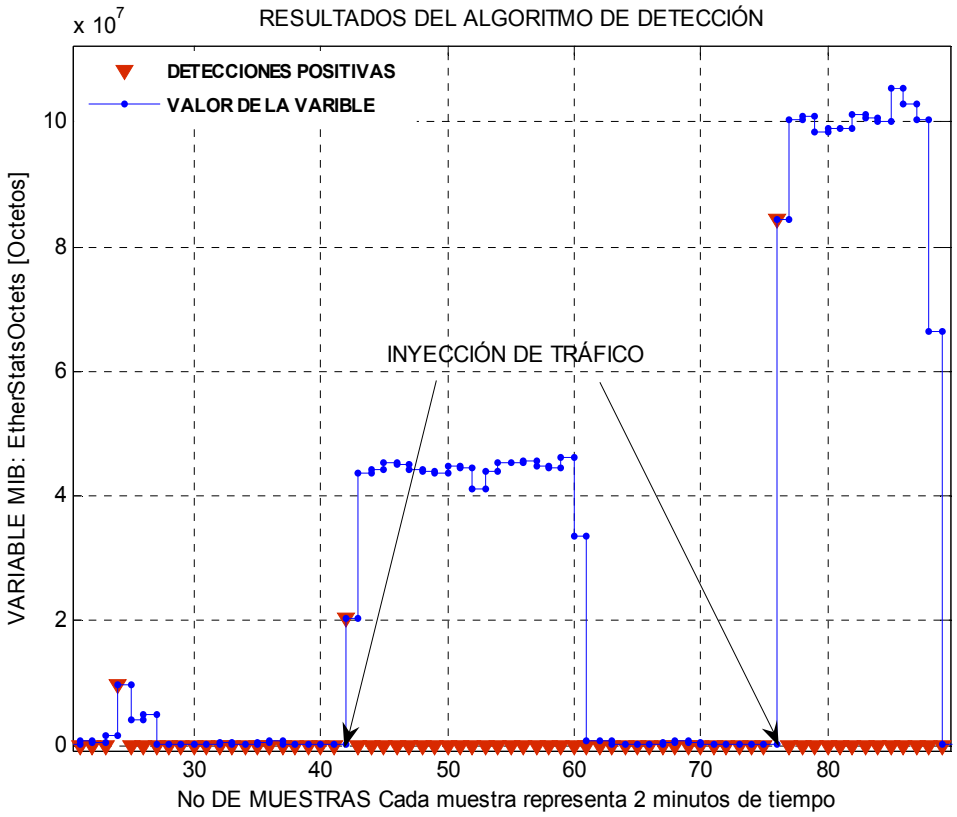
IfInUcastPkts e IfOutUcastPkts, para la falla de CONGESTIÓN					
	PORCENTAJE DE DETECCIONES DEL TOTAL DE MUESTRAS (Incluyendo falsas alarmas)	PROMEDIO DE DETECCIONES POR HORA	NÚMERO DE ANOMALÍAS DETECTADAS POR LA ENVOLVENTE	P_D*100	P_F*100
DETECTOR No1	4.06%	1.22	17/17	100%	2.95%
DETECTOR No2	1.26%	0,38	14/17	82.3%	0.33%
DETECTOR No3	4.46%	1.34	16/17	94.1%	3.42%

Figura 17. Comportamiento de las variables IfInUcastPkts y IfOutUcastPkts para la falla de congestión.



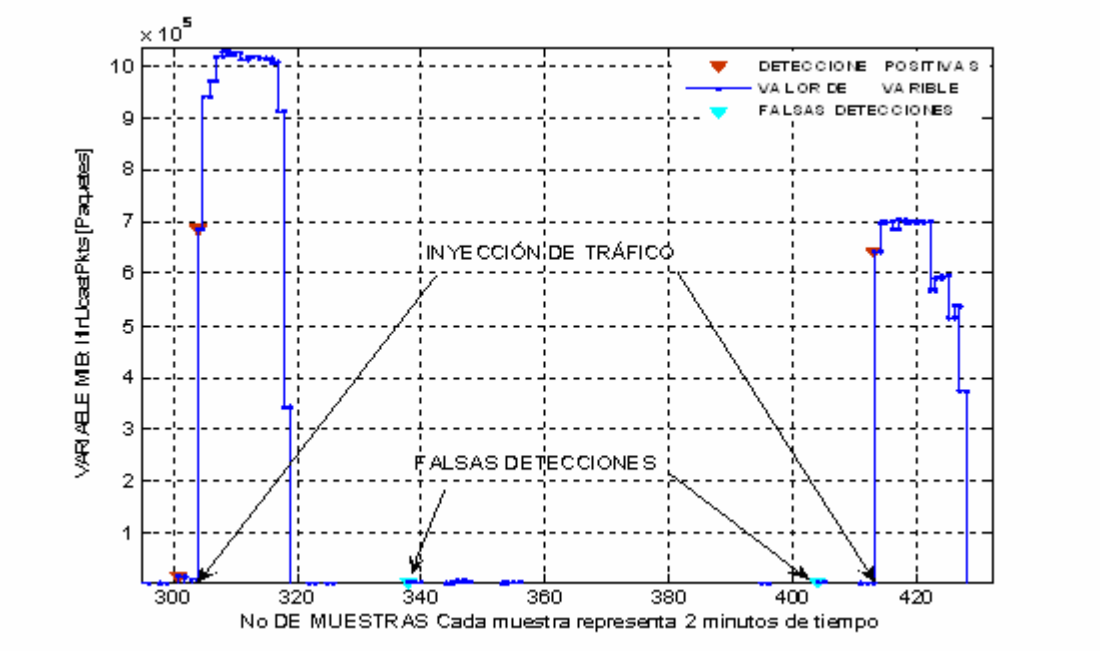
Una ventana de tiempo que muestra el comportamiento de la variable EtherStatsOctets y los momentos en que se presentan la detección de las anomalías obtenidas con el detector Nº 1 se muestra en la figura 17.

Figura 18. Comportamiento estadístico de la variable EtherStatsOctets y puntos de detección de anomalías obtenidos con el detector Nº 1 para 3 horas de monitorización.



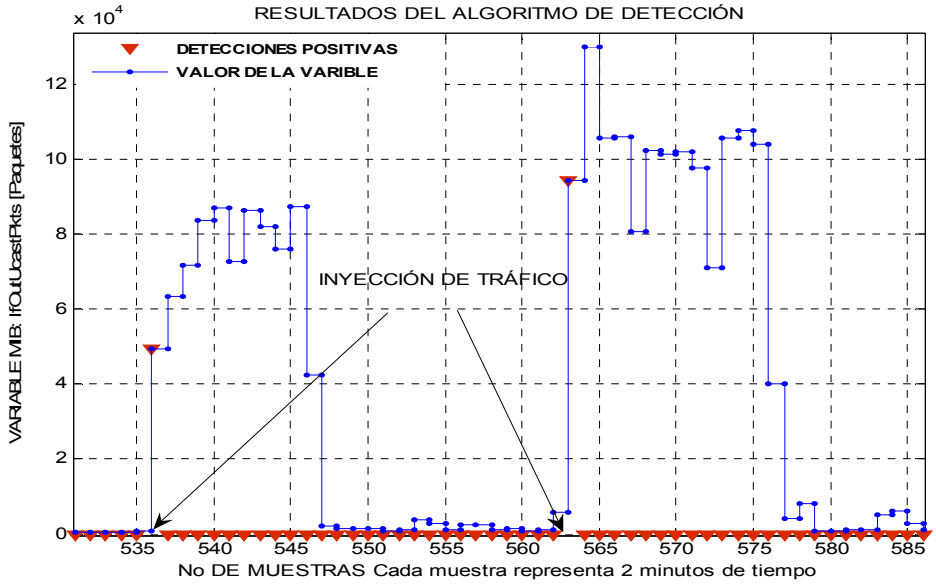
Una ventana de tiempo que muestra el comportamiento de la variable IfInUcastPkts y los momentos en que se presentan la detección de las anomalías obtenidas con el detector Nº 1 se muestra en la figura 18.

Figura 19. Comportamiento estadístico de la variable IfInUcastPkts y puntos de detección de anomalías obtenidos con el detector Nº 1 para 4.4 horas de monitorización.



Una ventana de tiempo que muestra el comportamiento de la variable IfOutUcastPkts y los momentos en que se presentan la detección de las anomalías obtenidas con el detector Nº 1 se muestra en la figura 19.

Figura 20. Comportamiento estadístico de la variable IfOutUcastPkts y puntos de detección de anomalías obtenidos con el detector Nº 1 para aproximadamente 2 horas de monitorización.



5.4.2 Resultados para la falla de Tormenta Broadcast.

- **Características de los datos:**

Nº total de muestras: 1052 muestras

Frecuencia de monitorización: 2 minutos

Tiempo total de sondeo =35,0666 horas

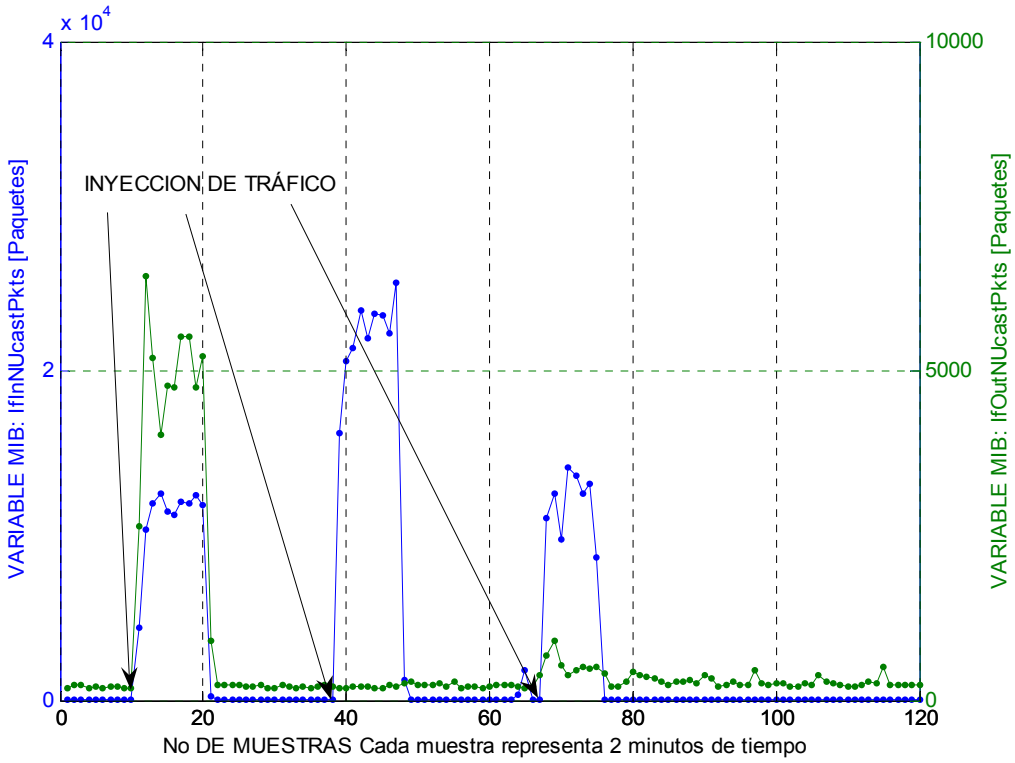
Número fallas de Tormenta de Broadcast generadas: 4

Con los resultados de las detecciones para las tres variables seleccionadas para esta falla se obtuvieron los resultados de la tabla 13; el comportamiento de este grupo de variables se observa en la figura 20; en el eje izquierdo se encuentra la variable que cuenta paquetes de difusión de entrada IfInNUcastPkts, y en el eje derecho la variable que cuenta paquetes de difusión de salida IfOutNUcastPkts.

Tabla 13. Resultados de detección en los tres algoritmos implementados para el grupo de variables de la falla de Tormenta de Broadcast.

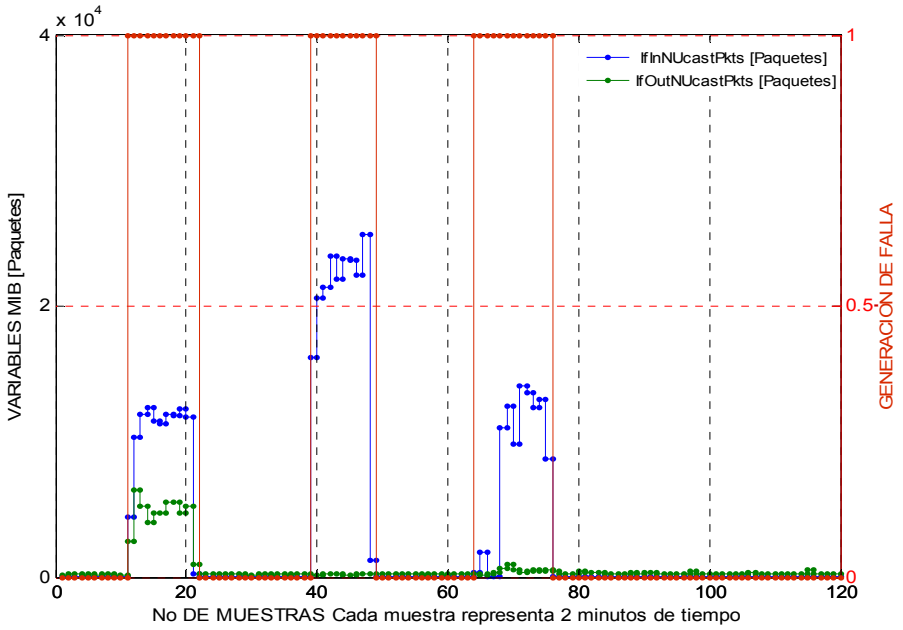
TORMENTA DE BROADCAST					
	PORCENTAJE DE DETECCIONES DEL TOTAL DE MUESTRAS (Incluyendo falsas alarmas)	**PROMEDIO DE DETECCIONES POR HORA**	**NÚMERO DE ANOMALÍAS DETECTADAS POR LA ENVOLVENTE**	**P_D*100**	**P_F*100**
DETECTOR No1	2.56%	0,769	4/4	100%	2.2%
DETECTOR No2	0.38%	0,114	3/4	75%	0.09%
DETECTOR No3	4.94%	1.482	4/4	100%	4.60%

Figura 21. Comportamiento del grupo de variables para la falla de Tormenta de Broadcast.



Como se esperaba, con estas dos variables se pueden detectar todas las fallas de Tormenta de broadcast generadas por medio de los detectores Nº 1 y Nº 3, como se aprecia en la tabla 13; en la figura 21 aparecen relacionadas estas variables con el momento de generación de la falla.

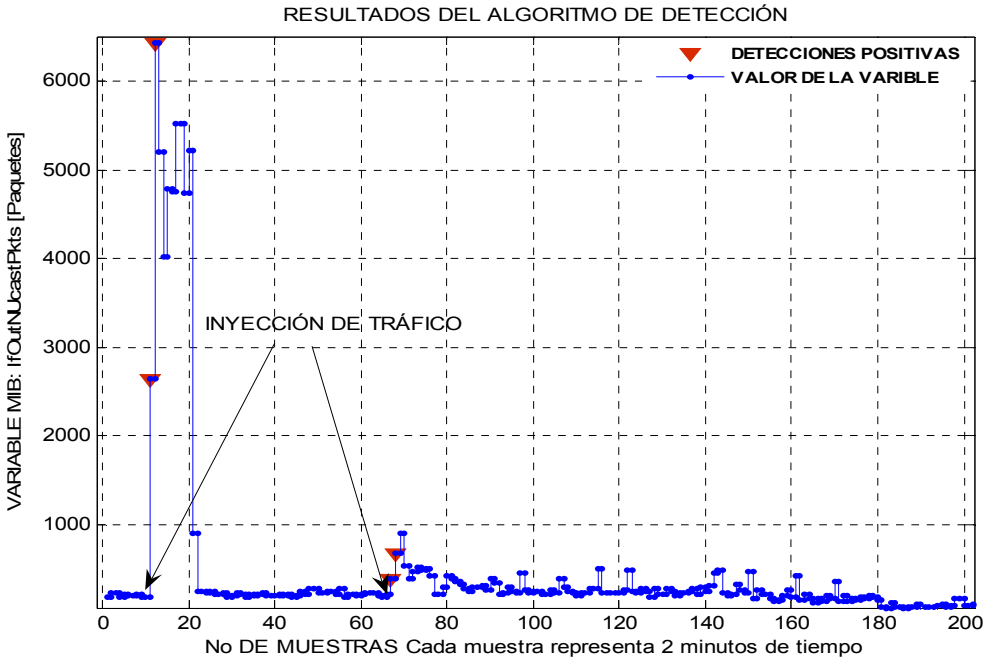
Figura 22. Comportamiento del grupo de variables para la falla de Tormenta de Broadcast.



Una ventana de tiempo que muestra el comportamiento de la variable IfInNUcastPkts y de la variable IfOutNUcastPkts y los momentos en que se presenta la detección de las anomalías obtenidas con el detector Nº 1 se muestra en las figuras 22 y 23 respectivamente.

Figura 23. Comportamiento estadístico de la variable IfInNUcastPkts y puntos de detección de anomalías obtenidos con el detector Nº 1 para 2.5 horas de monitorización.

Figura 24. Comportamiento estadístico de la variable IfOutNUcastPkts y puntos de detección de anomalías obtenidos con el detector Nº 1 para 6.6 horas de monitorización.



Los resultados obtenidos a partir de los algoritmos de detección implementados, muestran claramente la validez de la asociación entre las variables MIB seleccionadas y las condiciones específicas que dan origen a este tipo de falla.

Se puede apreciar la diferencia entre el número total de alarmas entre los grupos de variables asociados a las diferentes fallas. Las variables MIB asociadas a la falla de Congestión presentan una tasa de alarmas mayor (4.06 % y 3.26%) que las asociadas a Tormenta Broadcast (2.5%), debido a que en el grupo de congestión el comportamiento de las variables presenta más cambios, mientras el número de paquetes de broadcast tiende a mantenerse en un rango de amplitud de la variable más cerrado.

Los mejores resultados han sido logrados con el detector Nº 1 y por tal razón las gráficas presentadas para cada variable pertenecen al análisis de este algoritmo de detección. Con este algoritmo se detectaron todas las fallas generadas durante las pruebas, con una

probabilidad de falsas alarmas aceptable en cada variable en comparación con los demás algoritmos de detección implementados.

En oposición a esto se puede observar en las tablas de desempeño de los algoritmos, que el detector Nº 2 tiene una probabilidad de falsas alarmas muy baja, pero no alcanza a detectar algunas de las fallas generadas; 14 fallas de 17 para la falla de congestión y 3 fallas de 4 para la falla de tormenta de broadcast, lo que hace que la función primaria del detector no sea cumplida de manera satisfactoria. El detector Nº 3 solo deja de detectar una falla generada para el caso de la falla de congestión

A partir del desempeño mostrado por cada detector se puede observar la relación directa existente entre la probabilidad de falsas alarmas y la precisión en la detección de las fallas generadas, ya que una probabilidad alta de falsas alarmas indica un nivel elevado de sensibilidad del detector.

Esta característica debe ser analizada con mayor detalle, debido a que en cualquier sistema de detección de fallas se desea disminuir el número de falsas alarmas a un nivel mínimo, sin comprometer la detección precisa de las anomalías que están asociadas con condiciones de falla.

Como se observó en el análisis del grupo de variables seleccionadas para la falla de congestión a través del algoritmo de detección, solamente con la variable EtherStatsOctets se pueden obtener buenos resultados a la hora de indicar anomalías en la red relacionadas con esta falla. Aunque se puede tener la ventaja de que se pueda minimizar aún más el número de variables para detectar esta falla, se pierde algo de información relacionada con la falla, por ejemplo si seleccionamos las dos variables MIB que cuentan paquetes de unicast de entrada y de salida podríamos diferenciar la dirección del tráfico que origina la congestión de la red, mientras que con solo la variable EtherStatsOctets no hay información de donde provino el tráfico ni tampoco si corresponde exclusivamente a tráfico unicast. Sin embargo es una cuenta de bytes que atravesaron el canal durante el intervalo de muestreo

que representa muy bien una congestión. Además esta variable es la que más niveles de variación presenta y por lo tanto un detalle que se debe tener en cuenta, es que en algún momento pueda sobrepasar el límite de su contador[85] reiniciando el conteo. Esta observación también cuenta para la falla de Tormenta de broadcast ya que en algún momento se tuvo en cuenta la variable EtherStatsBroadcastPkts, la cual se descartó luego de que no fue soportada adecuadamente por el dispositivo.

5.5 COMPORTAMIENTO DE LAS VARIABLES AUXILIARES

Como se mencionó en el capítulo anterior, con la información obtenida de las variables auxiliares es posible complementar la información suministrada por las variables que se seleccionaron para cada falla.

Con la variable IfInErrors y la información de los paquetes de entrada a la interfaz se puede conocer el valor de la tasa de error a partir de la ecuación (6)[86].

$$\%\ Tasa\ de\ Error = \frac{IfInErrors * 100}{IfInUcastPkts + IfInNUcastPkts} \qquad (6)$$

Esta información nos da una medida del funcionamiento real del segmento de red en un período de tiempo dado. La aparición de errores en la interfaz es un síntoma de anormalidad de la red y una medida de esta anormalidad esta dada por la tasa de error la cual indica el porcentaje paquetes con errores del total de paquetes que se procesaron en el dispositivo en un intervalo de tiempo el cual corresponde a la frecuencia de monitorización de las variables MIB.

A partir de las pruebas realizadas para los dos tipos de fallas, se puede visualizar que 3 de estas presentaron un alto nivel de severidad, representado por el aumento considerable de la

[85] Ver anexo B, Counter32.

[86] BLUM, Richard. NETWORK PERFORMANCE OPEN SOURCE TOOLKIT, Wiley Publishing, Inc. 2003. Capitulo 3.

tasa de error como se aprecia a partir de las figuras 25, 26 ,27 y 28.

Figura 25. Comportamiento de la Tasa de error en las pruebas realizadas en la falla de congestión.

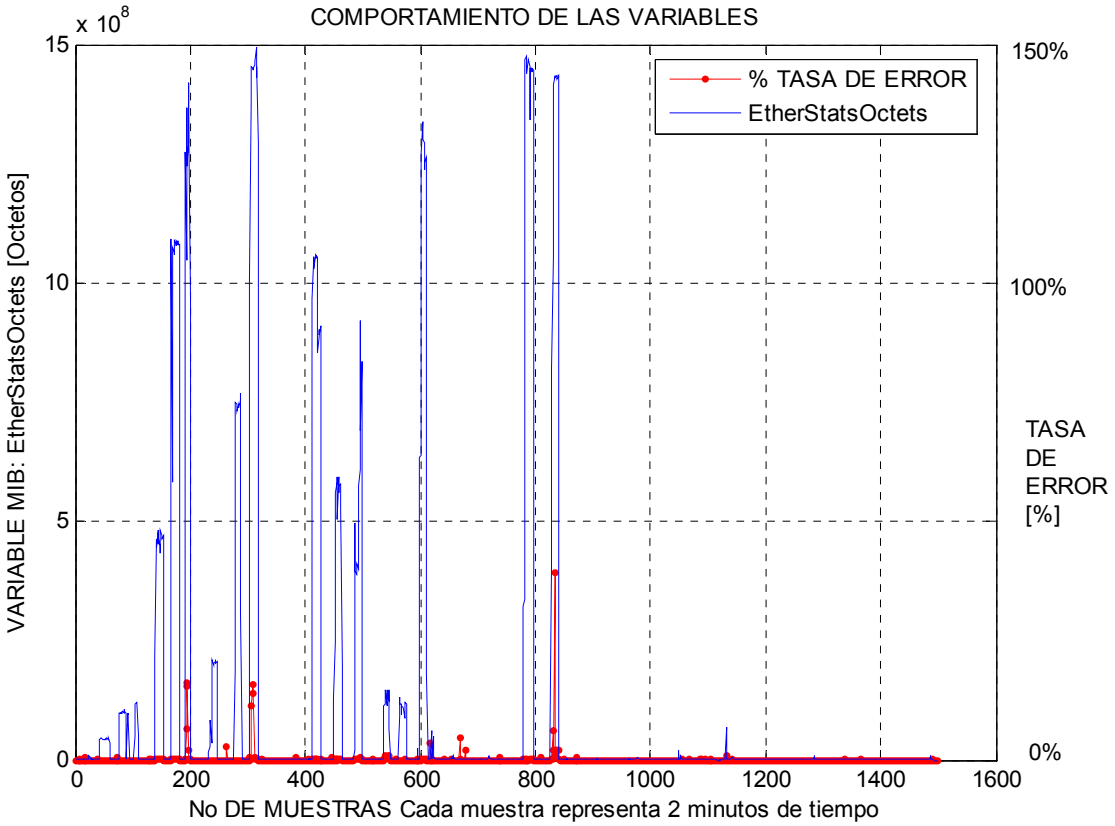


Figura 26. Comportamiento de la Tasa de error y de la variable EtherStatsCollisions en las pruebas realizadas en la falla de congestión.

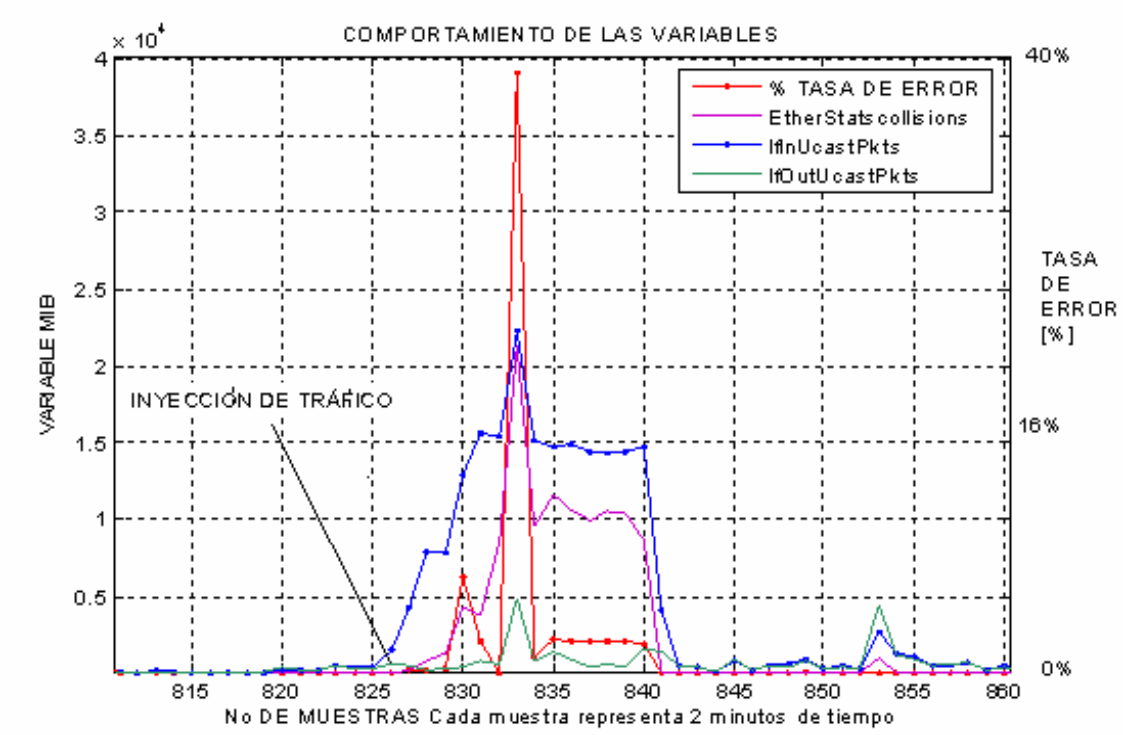


Figura 27. Comportamiento de la Tasa de error en las pruebas realizadas en la falla de congestión.

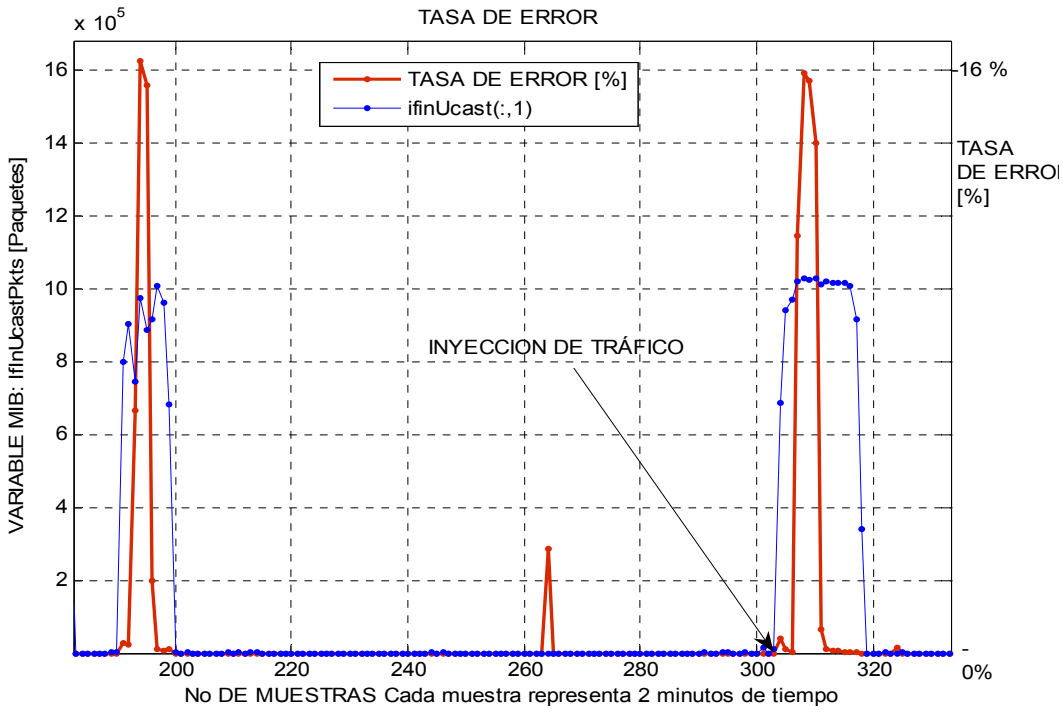
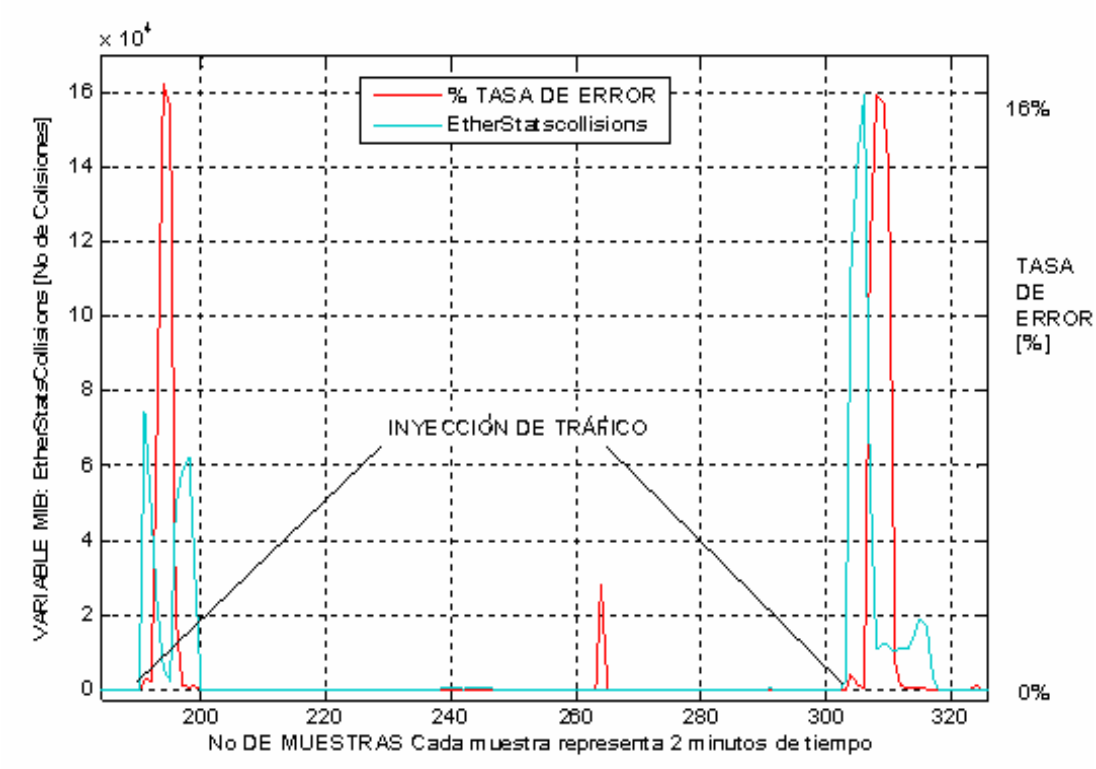


Figura 28. Comportamiento de la variable EtherStatsCollisions y de la Tasa de Error.



En la figura 26, se puede apreciar fácilmente que la tasa de error se incrementa unas muestras después de que se detecte la anomalía, lo cual muestra claramente la utilidad de medir la tasa de error, como método alterno de análisis del funcionamiento de la red.

En la figura 27 se observa que, como era de esperarse, se presentó un gran aumento en el valor de la variable EtherStatsCollisions al estar congestionado el segmento de red, lo cual dispara inmediatamente el número de colisiones presentes en el segmento de prueba; también se aprecia el aumento de errores con una leve disminución de las colisiones en el momento de la falla.

De las variables auxiliares seleccionadas, se observó que la variable IfInDiscards no presentó variación en todo el tiempo de las pruebas; por lo que se resalta la robustez, capacidad y calidad del agente estudiado.

6. CONCLUSIONES

La notoria ausencia en la literatura científica de procedimientos prácticos y generales a la hora de diseñar e implementar metodologías de gestión de redes automatizadas, basadas en la información obtenida de los dispositivos que hacen parte de ellas, hacen significativa la contribución de este trabajo. Esta situación es relevante debido a que el planteamiento de los criterios y procedimientos hechos de manera detallada en este trabajo es omitido con frecuencia en la bibliografía existente, lo cual dificulta el establecimiento de un consenso general para la creación de estas herramientas de gestión. En resumen, la temática tratada en este documento es la base y punto de partida para cualquiera de estos desarrollos.

Después de un análisis profundo de los resultados conseguidos en esta investigación, se mencionan los más representativos a continuación.

- La alta variabilidad de las condiciones de tráfico de las redes y la carencia de modelos precisos que ayuden a describirlas, sumada a la heterogeneidad de la información sobre el tráfico que puede ser obtenida de la propia red, gracias a su naturaleza estocástica[87]; provoca que la adecuada selección de un grupo de variables MIB sea crítica. Por lo tanto todo proceso de selección de variables MIB debe orientarse a la eficiencia y a obtener la máxima información del menor número de variables posible; ya que de lo contrario las tareas de gestión serían totalmente inconvenientes debido a que estas son por definición altamente intrusivas, y por lo tanto generadoras de congestión.

Los factores clave en la selección de variables son la disponibilidad de los grupos estándar de la MIB en la configuración básica de los dispositivos gestionables con los que se cuente,

[87] Se denomina estocástico a aquel sistema que funciona, sobre todo, por el azar. Las leyes de causa-efecto no explican cómo actúa de manera determinista, sino en función de probabilidades.

el nivel OSI en el que operan y que la configuración desplegada habilite todas las opciones de monitorización. La generalidad de las definiciones estándar y la abundancia de las variables, genera una alta redundancia de funciones entre grupos de variables de diferente orientación que obligan a definir de manera individual los criterios bajo los que se realiza la discriminación entre las diferentes variables.

- Las tablas[88] de evaluación de desempeño expuestas para cada algoritmo de detección implementado, muestran que es inevitable incurrir en falsas detecciones si se desea cierto nivel de precisión para detectar el momento en que se induce la falla; el detector Nº 1 logro detectar todas las pruebas realizadas evidenciando este comportamiento anómalo para cada variable en presencia de falla.

- Se observó que cuando una falla se presenta con gran intensidad como lo muestran las figuras 25 y 26, la tasa de error comienza a incrementarse dentro de un rango de tiempo de 3 a 6 minutos después de iniciada la generación de tráfico y de efectuarse la detección de alguna anomalía mediante el algoritmo de detección; esto indica que con la monitorización de las variables MIB seleccionadas, se puede implementar un sistema de detección proactivo que sea capaz de informar al administrador de la red, antes de que los problemas originados por la presencia de la falla sean severos; con esto se demuestra una de las grandes utilidades de la información que se obtiene a partir de la monitorización de las variables MIB.

- Los criterios empleados en la clasificación de variables y el análisis realizado en el proceso de detección, pueden cambiar dependiendo de la falla presente en la red y de las condiciones asociadas específicamente a cada una de ellas, así como de los dispositivos gestionables usados y el tamaño de la red. Por esta razón el propósito de este trabajo se centró en la metodología representada en el desarrollo de este proyecto.

[88] Ver tablas 10 -13

7. RECOMENDACIONES

La profundidad y complejidad de la temática tratada, plantea varios interrogantes que están fuera del alcance de esta investigación, pero que debido a lo innovador y poco explorado de esta área de la gestión de redes, puede dar origen a proyectos que busquen su solución.

A continuación se mencionan algunos aspectos que podrían motivar investigaciones posteriores.

- El paso a seguir después de la puesta en práctica de la metodología seguida en este trabajo es la creación de una herramienta de gestión o de detección y clasificación de fallas basada en las Variables MIB, que funcione en tiempo real. Para este propósito se sugiere la implementación de algoritmos que hagan uso de alguna aplicación de la inteligencia artificial, como las redes neuro-fuzzy, sistemas expertos, agentes inteligentes, etc.

- Se recomienda aplicar el procedimiento seguido en este estudio en una red de mayor tamaño, por ejemplo la red institucional de la Universidad Industrial de Santander, con el fin de establecer la base para el desarrollo de un sistema de gestión de redes enfocado a las características y necesidades específicas de la misma.

- Aprovechando el gran impulso e interés que despierta actualmente el estudio de las redes inalámbricas WLAN al interior del grupo de investigación en conectividad y procesado de señal (CPS), de la escuela de ingenierías Eléctrica, Electrónica y de Telecomunicaciones, se recomienda realizar una investigación para definir la aplicabilidad de los criterios y conceptos expuestos en este trabajo al campo de las redes inalámbricas, y de esta forma promover el desarrollo de herramientas de gestión en esta clase de redes.

BIBLIOGRAFÍA

BLUM, Richard. NETWORK PERFORMANCE OPEN SOURCE TOOLKIT, Wiley Publishing, Inc. 2003 ISBN: 0-471-43301-2. Libro de referencia sobre pruebas de desempeño de redes mediante la aplicación de las herramientas software más comunes.

BREITBART, Yuri; CHAN, Chee-Yong; GAROFALAKIS, Minos; RASTOGI, Rajeev; SILBERSCHATZ, Avi. "Efficiently Monitoring Bandwidth and Latency in IP Networks", Information Sciences Research Center, Bell Laboratories 2000. Este documento plantea diversos métodos para realizar de manera eficiente procesos de monitoreo de parámetros de desempeño en redes basadas en IP.

BREKNE, Tønnes; CLEMETSEN, Marius; HEEGAARD, Poul; INGVALDSEN, Tone; VIKEN, Brynjar. STATE OF THE ART IN PERFORMANCE MONITORING AND MEASUREMENTS, Telenor R&D R 15/2002 ISBN 82-423-0530.5. Este libro ofrece una introducción a las medidas de desempeño de redes IP, y describe los conceptos y métodos en uso actualmente para el estudio del desempeño de redes, sus actores y herramientas.

CABRERA, J; LEWIS, L; QIN, X; LEE, W; PRASANTH, R; RAVICHANDRAN, B; MEHRA, R. "Proactive Detection of Distributed Denial of Service Attacks using MIB Traffic Variables a Feasibility Study", IEEE 2001. Este documento muestra una propuesta de metodología para la utilización de sistemas de gestión de red (NMS) para la detección temprana de ataques distribuidos de negación de servicio.

CARRASCO, J; CAMPOS, J; RUIZ, C. "Gestión De Red: Protocolo SNMP" Grupo de Investigación en Señales, Telemática y Comunicaciones, GSTC. Universidad de Granada España, http://ceres.ugr.es/ 2004. Presenta información general sobre gestión de redes, el protocolo SNMP y sus aplicaciones.

CHAO, C.S; YANG D.L; LIU A.C. “An Automated Fault Diagnosis System Using Hierarchical Reasoning and Alarm Correlation”, Feng Chia University,Taiwán 2000. En este documento se presenta el desarrollo de un sistema práctico para el diagnostico de fallas de red.

CISCO SYSTEMS. INTERNETWORKING TECHNOLOGY HANDBOOK, Capítulos 2, 6, 55, 56 2004. Disponible en Internet: http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/. Este documento presenta información general sobre gestión de redes, protocolo SNMP, monitoreo remoto RMON, etc.

DELLA MAGIORA Paul; ELLIOT, Christopher. PERFORMANCE AND FAULT MANAGEMENT, Cisco press 2000 ISBN: 1-57870-180-5. Este Libro es una referencia en las áreas de medidas de desempeño de redes y gestión de fallas.

DUARTE , Elías Procópio; DOS SANTOS, Aldri L. “Network Fault Management Based on SNMP Agent Groups”, IEEE 2001. En este documento se presenta una estructura novedosa para la monitorización de objetos de gestión SNMP en una red de área local, se describe la creación e implementación de una herramienta para la gestión de fallas de red en una LAN.

DUBUISSON, Olivier. ASN.1 COMMUNICATION BETWEEN HETEROGENEOUS SYSTEMS, OSS Nokalva, 2000 ISBN: 0-12-6333361-0. Este libro presenta la teoría sobre la notación de sintaxis abstracta 1.

FEATHER, Frank; SIEWIOREK, Dan; MAXION, Roy. “Fault Detection in an Ethernet Network Using Anomaly Signature Matching”, SIGCOMM 1993. Se presenta un método para la detección automática de problemas en una red de área local Ethernet, para lo cual se hace una investigación profunda sobre los tipos de condiciones de falla que afectan a esta clase de redes.

HOOD, Cynthia S; JI, Chuanyi**.** Intelligent Agents For Proactive Fault Detection, IEEE Internet Computing, March -April 1998. En este documento se presenta la creación de una herramienta de detección de fallas, basada en agentes inteligentes.

HIGBIE, Carrie. "Congestion-Can standards provide relief?", The Siemon Company 2004. Este documento presenta conceptos y definiciones sobre el estado de congestión en una red.

LAN TRAFFIC V2–USER GUIDE, ZTI 1998-2003. All rights reserved. France Telecom licensed product. www.zti-telecom.com. Esta fuente presenta el manual de usuario del software LAN Traffic.

MARABOLI ROSSELOTT, Marcelo. "Monitor de Tráfico Ethernet Netgraph" Memoria de grado, Universidad Técnica Federico Santa María de chile, Departamento de Electrónica Noviembre 1997. Trabajo de investigación a nivel de pregrado relacionado con el área de gestión de redes haciendo un énfasis especial en el estudio de herramientas de monitorización y sus propiedades.

MONTGOMERY, Douglas C; RUNGER, George C. PROBABILIDAD Y ESTADÍSTICA APLICADAS A LA INGENIERÍA, Segunda Edición, Editorial Limusa S.A. 2002. Libro de estadística general con aplicaciones especificas en el campo de la ingeniería.

OPPENHEIM, Alan V; WILLSKY, Alan S. SEÑALES Y SISTEMAS. 2da ed., Prentice Hall Hispanoamericana, 1998. Libro de referencia sobre procesamiento de señales y análisis de sistemas.

RFC´S 1156, 1158, 1212, 1213, 1271, 1724, 1757, 2914. En esta serie de notas sobre el Internet, se presentan documentos que contienen proposiciones, comentarios y los estándares relacionados a la tecnología del Internet, específicamente sobre SNMP, la MIB y la Gestión de Redes propuesta por el IETF.

SAYENKO, Oleksandr. “Policy Based Model for Monitoring SNMP Resources”, Master’s Thesis Work, University of Jyväskylä, Finland, Department of Mathematical Information Technologies 8/8/2002. Trabajo de investigación a nivel de maestría relacionado con el área de gestión y monitorización de redes.

SPURGEON, Charles. ETHERNET: THE DEFINITIVE GUIDE, O’Reilly & Associates, 2000. ISBN 1-56592-660-9. Tomado como libro de consulta referente a conceptos de Ethernet, en especial el capítulo 19 que trata sobre el desempeño de redes.

STALLINGS William. COMUNICACIONES Y REDES DE COMPUTADORES, Sexta Edición, Pearson Educación S.A.: 2000. Libro de referencia general sobre redes, con descripción de protocolos y sistemas de referencia.

STALLINGS William. SNMP, SNMPV2, SNMPV3, AND RMON1 AND 2, Addison-Wesley, ISBN: 0201485346 1999. Libro de estudios especifico sobre el protocolo SNMP en sus diferentes versiones y su aplicación a la gestión de redes.

STEVENS W.R. TCP/IP ILLUSTRATED, Vol. 1: THE PROTOCOLS, Addison- Wesley 1994. Libro de consulta sobre los protocolos de redes en especial el protocolo SNMP.

SUBRAMANIAN M. NETWORK MANAGEMENT- PRINCIPLES AND PRACTICE, Addison-Wesley, 2000 ISBN: 0201357429. Este libro profundiza en el área de gestión de redes y es de importancia especial el capítulo 13, donde se trata la gestión de fallas.

TANENBAUN Andrew S. REDES DE COMPUTADORAS, Tercera Edición, Prentice Hall Hispanoamericana S.A., 1997. Libro de consulta sobre información general de redes de computadoras, presenta un capitulo dedicado al desempeño de redes.

THOTTAN, M; JI, Chuanyi. “Adaptive Thresholding for Proactive Network Problem Detection”, Rensselaer Polytechnic Institute 1999. Este documento presenta un estudio encaminado a detectar problemas de red potenciales a través de mediciones del tráfico de red, utilizando como referencia a las variables de la MIB.

THOTTAN, M; JI, Chuanyi. “Anomaly Detection in IP Networks”, IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL 51, NO. 8, AUGUST 2003. Este documento presenta conceptos sobre las anomalías más frecuentes en redes Ethernet y las formas de detectarlas.

THOTTAN, M. “MIB Variable Based Fault Classification: The Next Step Towards Proactive Management”, Bell Labs 2001. Se presenta un estudio en el que se plantea una clasificación de fallas de red, utilizando para ello la información contenida en las variables MIB.

THOTTAN, M; JI, Chuanyi. “Proactive Anomaly Detection Using Distributed Intelligent Agents”, IEEE Network September/ October 0890-8044/ 1998. Este documento presenta un procedimiento para la detección proactiva de fallas de red mediante la aplicación de agentes inteligentes.

THOTTAN, M; JI, Chuanyi. “Properties of Network Faults”, Bell Labs 2001. Este documento presenta una descripción de las fallas de red más comunes y sus características.

THOTTAN, M; JI, Chuanyi. “Statistical Detection of Enterprise Network Problems”, Rensselaer Polytechnic Institute 1999. Este documento presenta un procedimiento para la detección de problemas de red basándose en métodos estadísticos.

WAN KILLER OVERVIEW. Wan Killer Network traffic generator Help. Copyright 1995-2002. www.solarwinds.net. Esta fuente presenta el archivo de ayuda de la herramienta de generación Wan Killer que hace parte del paquete Solarwinds.

YUMOTO, K. “What's TFGEN”. Copyright 1996. www.st.rim.or.jp/~yumo/index.html. Esta fuente presenta el manual de usuario para el software TFGEN.

ANEXO A. Tablas de Variables MIB.

A continuación se presentan las tablas de variables de la MIB seleccionadas inicialmente para su estudio y clasificación, teniendo en cuenta su posible asociación a las fallas bajo estudio. Este conjunto de variables hacen parte de los grupos estándar de la MIB II.

System (1.3.6.1.2.1.1)

NOMBRE	OID	SINTAXIS	ENUMERACIÓN	DESCRIPCIÓN
sysUpTime	1.3.6.1.2.1.1.3	TimeTicks		Indica la cantidad de tiempo desde que el sistema fue reiniciado por última vez en centésimas de segundo.

Interfaces (1.3.6.1.2.1.2)

NOMBRE	OID	SINTAXIS	ENUMERACIÓN	DESCRIPCIÓN
ifNumber	1.3.6.1.2.1.2.1	Integer32		Registra el número total de interfaces de red, independientemente de su estado actual.
ifDescr	1.3.6.1.2.1.2.2.1.2	Octet String	DisplayString	Muestra información textual acerca de la interfaz. Debe incluir el nombre del fabricante, del producto y la versión del hardware y el software de la interfaz.
ifType	1.3.6.1.2.1.2.2.1.3	Integer32	1-161	Muestra el tipo de interfaz; valores adicionales de esta variable son asignados por la IANA.
ifAdminStatus	1.3.6.1.2.1.2.2.1.7	Integer32	up 1 down 2 testing 3	Muestra el estado deseado de la interfaz. El estado de prueba (3) indica que no pueden pasar paquetes operacionales. Cuando un sistema gestionado inicializa, todas las interfases arrancan con este objeto en el estado bajo (2). Como resultado de una acción de gestión explicita o por información de configuración almacenada por el sistema gestionado, el objeto es cambiado a los estados activo (1) o prueba (3).

ifOperStatus	1.3.6.1.2.1.2.2.1.8	Integer32	up 1 down 2 testing 3 unknown 4 dormant 5 notPresent 6 lowerLayer Down 7	Muestra el estado operacional actual de la interfaz. El estado prueba (3) indica que no pueden ser pasados paquetes operacionales. Si ifAdminStatus esta bajo (2) entonces este objeto debe estar bajo. Si ifAdminStatus es cambiado a activo entonces este objeto debe cambiar a arriba(1), Si la interfaz esta lista para transmitir y recibir tráfico de la red; si la interfaz esta esperando por acciones externas (como una línea serial esperando por una conexión entrante) debe cambiar al estado durmiendo (5); debe permanecer en el estado bajo (2) si y solo si existe una falla que la previene de cambiar al estado arriba (1); debe permanecer en estado No presente (6) si la interfaz tiene componentes, generalmente de hardware inactivos
IfInOctets	1.3.6.1.2.1.2.2.1.10	Counter32		Muestra el número total de objetos recibidos en la interfaz, incluyendo caracteres de trama.
IfInUcastPkts	1.3.6.1.2.1.2.2.1.11	Counter32		Muestra el número de paquetes entregados por esta subcapa a una superior, los cuales no fueron direccionados a una dirección Broadcast ni multicast en esta subcapa.
ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12	Counter32		Muestra el número de paquetes entregados por esta subcapa a una superior, los cuales fueron direccionados a una dirección Broadcast o multicast en esta subcapa.
ifInErrors	1.3.6.1.2.1.2.2.1.14	Counter32		Para interfaces orientadas a paquetes, muestra el número de paquetes de entrada que contienen errores, evitando su entrega a protocolos de capa superior. Para interfaces orientadas a caracteres o de longitud fija, muestra el número de unidades de transmisión de entrada que contienen errores, evitando su entrega a protocolos de capa superior.

IfInUnknown Protos	1.3.6.1.2.1.2.2.1.15	Counter32		Para interfaces orientadas a paquetes, muestra el número de paquetes recibidos por la interfaz que fueron descartados debido a un protocolo desconocido o no soportado. Para interfaces orientadas a caracteres o de longitud fija que soporten protocolos multiplexados, muestra el número de unidades de transmisión recibidas a través de la interfaz que fueron descartadas debido a un protocolo desconocido o no soportado. Para cualquier interfaz que no soporte protocolos multiplexados este contador será siempre 0.
ifOutOctets	1.3.6.1.2.1.2.2.1.16	Counter32		Muestra el número total de octetos transmitidos fuera de la interfaz incluyendo caracteres de tramas.

Ip (1.3.6.1.2.1.4)

NOMBRE	OID	SINTAXIS	ENUMERACIÓN	DESCRIPCIÓN
IpInReceives	1.3.6.1.2.1.4.3	Counter32		Indica el número total de datagramas de entrada recibidos desde las interfaces, incluyendo aquellos recibidos con errores.
IpInHdrErrors	1.3.6.1.2.1.4.4	Counter32		Indica el número de datagramas de entrada descartados debido a errores en sus cabeceras IP, incluyendo malas sumas de comprobación, números de versión incongruentes, otros errores de formato, exceso del time to live, errores descubiertos en el procesamiento de sus opciones IP, etc.
IpInAddrErrors	1.3.6.1.2.1.4.5	Counter32	DisplayString	Indica el número de datagramas de entrada descartados debido a que las direcciones IP en los campos de destino de sus cabeceras IP no son direcciones validas para ser recibidas en esta

				entidad. Este conteo incluye direcciones invalidas (0.0.0.0) y direcciones de clases no soportadas (clase E). Para entidades que no son routers IP y por eso no reenvían datagramas, este contador incluye datagramas descartados debido a que la dirección de destino no es una dirección local.
IpForwDatagrams	1.3.6.1.2.1.4.6	Counter32		Indica el número de datagramas de entrada para los que esta entidad no fue su destino IP final, como resultado de esto se hizo el intento de encontrar una ruta para reenviarlos a su destino final. En entidades que no actúan como routers IP, este contador incluirá solo aquellos paquetes que fueron enrutados desde la fuente a través de esta entidad, y el procesamiento de la opción de enrutamiento desde la fuente fue exitoso.
IpInUnknown Protos	1.3.6.1.2.1.4.7	Counter32		Indica el número de datagramas direccionados localmente recibidos satisfactoriamente pero descartados debido a un protocolo desconocido o no soportado
IpInDiscards	1.3.6.1.2.1.4.8	Counter32		Indica el número de datagramas IP de entrada para los que no hubo problemas al prevenir su procesamiento constante, pero que fueron descartados (por falta de espacio en los buffer). Este contador no incluye ningún datagrama descartado mientras espera su reensamble. Si el ancho de banda de la interfaz es mayor que el máximo valor para este objeto (4,294,967,295), luego se debe utilizar el objeto ifHighspeed para reportar la velocidad de esta interfaz, para una subcapa a la cual no se aplique el concepto de ancho de banda este valor podría ser cero.

IpInDelivers	1.3.6.1.2.1.4.9	Counter32		Indica el número total de datagramas de entrada entregados satisfactoriamente a los protocolos IP de usuario (incluyendo ICMP).
IpOutRequests	1.3.6.1.2.1.4.10	Counter32		Indica el número total de datagramas cuyos protocolos IP de usuarios locales (incluyendo ICMP) suplieron las peticiones de IP para transmisión. Este contador no incluye ningún datagrama contado en ipForwDatagrams.
IpOutDiscards	1.3.6.1.2.1.4.11	Counter32		Indica el número de datagramas de entrada para los que no hubo problemas al prevenir su transmisión a su destino, pero que fueron descartados (por falta de espacio en los buffer). Este contador podría incluir datagramas contados en ipForwDatagrams si cualquiera de esos paquetes cumple este (estricto) criterio de descarte.
IpOutNoRoutes	1.3.6.1.2.1.4.12	Counter32		Indica el número de datagramas IP descartados debido a que no se encontró ruta para transmitirlos a su destino. Este contador incluye cualquier paquete contado en ipForwDatagrams que cumpla este criterio de no ruta. Esto también incluye cualquier datagrama cuyo Host no pueda ser enrutado debido a que sus routers por defecto están fuera de línea.
IpReasmReqds	1.3.6.1.2.1.4.14	Counter32		Indica el número de fragmentos IP recibidos en esta entidad que necesitan ser reensamblados.

IpReasmFails	1.3.6.1.2.1.4.16	Counter32		Indica el número de fallas detectadas por el algoritmo de reensamble IP (por cualquier razón: Tiempo agotado, errores, etc.). No es necesaria una cuenta de fragmentos IP descartados desde que algunos algoritmos (en especial el algoritmo RFC 815) pueden perder la pista del número de fragmentos debido a la combinación de ellos a medida que son recibidos.
IpFragCreates	1.3.6.1.2.1.4.19	Counter32		Indica el número de fragmentos de datagramas IP que han sido generados como resultado de la fragmentación en esta entidad.
IpRouteType	1.3.6.1.2.1.4.21.1.8	Integer32	other 1 invalid 2 direct 3 indirect 4	Muestra el tipo de ruta. Los valores directo (3) e indirecto (4) se refieren a la noción de enrutamiento directo e indirecto en la arquitectura IP. Fijar este objeto con el valor inválido (2) tiene el efecto de invalidar la entrada correspondiente en la variable ipRouteTable. Así efectivamente se desvincula el destino identificado con dicha entrada de la ruta identificada con dicha entrada. Esta es una implementación específica tan importante como que el agente remueva una entrada inválida de la tabla. De acuerdo a esto las estaciones de gestión deben ser preparadas para recibir información tabulada desde los agentes que corresponden a entradas que no están en uso actualmente. La interpretación adecuada de tales entradas requiere el examen de la variable ipRouteType apropiado.
IpRouting Discards	1.3.6.1.2.1.4.23	Counter32		Muestra el número de entradas de enrutamiento que fueron escogidas para ser descartadas aun siendo validas. Una razón para descartar tales entradas

				podría ser para liberar espacio en los buffer para otras entradas de enrutamiento.
ipCidrRoute Status	1.3.6.1.2.1.4.24.4.1.16	Integer32	active 1 notInService 2 notReady 3 createAndGo 4 createAnd Wait 5 destroy 6	Indica la variable de estado de la fila, usada de acuerdo a las normas de instalación y eliminación de filas.

Icmp (1.3.6.1.2.1.5)

NOMBRE	OID	SINTAXIS	ENUMERACIÓN	DESCRIPCIÓN
icmpInMsgs	1.3.6.1.2.1.5.1	Counter32		Indica el número total de mensajes ICMP que fueron recibidos por esta entidad. Este contador incluye todos los valores contados en la variable icmpInErrors.
icmpInEchos	1.3.6.1.2.1.5.8	Counter32		Indica él número de mensajes ICMP de eco (petición) recibidos.
icmpInEchoReps	1.3.6.1.2.1.5.9	Counter32		Indica el número de mensajes ICMP de respuesta de ecos recibidos.

TCP (1.3.6.1.2.1.6)

NOMBRE	OID	SINTAXIS	ENUMERACIÓN	DESCRIPCIÓN
tcpActiveOpens	1.3.6.1.2.1.6.5	Counter32		Indica el número de veces que las conexiones TCP han hecho una transición directa al estado SYN-SENT a partir del estado CLOSED (cerrado).
tcpPassiveOpens	1.3.6.1.2.1.6.6	Counter32		Indica el número de veces que las conexiones TCP han hecho una transición directa al estado SYN-RCVD a partir del estado LISTEN.
tcpCurrEstab	1.3.6.1.2.1.6.9	Unsigned32		Indica el número de conexiones TCP para las cuales el estado actual es ESTABLISHED o CLOSE-WAIT.
tcpInSegs	1.3.6.1.2.1.6.10	Counter32		Indica el número total de segmentos recibidos, incluyendo aquellos recibidos en error. Esta cuenta incluye

				los segmentos recibidos en las conexiones actuales
tcpInErrs	1.3.6.1.2.1.6.14	Counter32		Indica el número total de segmentos recibidos en error (malas sumas de verificación TCP, etc.)

UDP (1.3.6.1.2.1.7)

NOMBRE	OID	SINTAXIS	ENUMERACIÓN	DESCRIPCIÓN
UdpInDatagrams	1.3.6.1.2.1.7.1	Counter32		Indica el número total de datagramas UDP entregados a usuarios UDP.
UdpOutDatagrams	1.3.6.1.2.1.7.4	Counter32		Indica el número total de datagramas UDP enviados desde esta entidad.

Transmission (1.3.6.1.2.1.10)

NOMBRE	OID	SINTAXIS	ENUMERACIÓN	DESCRIPCIÓN
dot3Stats SingleCollision Frames	1.3.6.1.2.1.10.7.2.1.4	Counter32		Muestra una cuenta de las tramas transmitidas satisfactoriamente sobre una interfaz en particular para la cual la transmisión es frenada exactamente por una sola colisión. Una trama que es contada por una instancia de esta variable, también es contada por la instancia correspondiente de las variables ifOutUcastPkts, ifOutMulticastPkts, o ifOutBroadcastPkts; y no es contada por la instancia correspondiente de la variable dot3StatsMultipleCollisionFrames. Este contador no se incrementa cuando la interfaz esta operando en modo full-duplex
dot3Stats MultipleCollision Frames	1.3.6.1.2.1.10.7.2.1.5	Counter32		Muestra una cuenta de las tramas transmitidas satisfactoriamente sobre una interfaz en particular para la cual la transmisión es frenada por más de una colisión. Una

				trama que es contada por una instancia de esta variable, también es contada por la instancia correspondiente de las variables ifOutUcastPkts, ifOutMulticastPkts, o ifOutBroadcastPkts; y no es contada por la instancia correspondiente de la variable dot3StatsSingleCollisionFrames. Este contador no se incrementa cuando la interfaz esta operando en modo full-duplex.
dot3Stats SQETestErrors	1.3.6.1.2.1.10.7.2.1.6	Counter32		Muestra una cuenta de las veces que el mensaje SQE TEST ERROR es generado por la subcapa PLS (physical layer signaling) para una interfaz en particular. El SQE TEST ERROR es fijado de acuerdo a las reglas de verificación del mecanismo de detección SQE (signal quality error) en la función de detección de portadora PLS como se describe en IEEE Std. 802.3, 1998 Edición, sección 7.2.4.6. Este contador no se incrementa en interfaces que operan a velocidades mayores a 10Mb/s, o en interfaces que están operando en modo full-duplex.
dot3Stats Deferred Transmissions	1.3.6.1.2.1.10.7.2.1.7	Counter32		Muestra una cuenta de tramas para las cuales el primer intento de transmisión sobre una interfaz en particular es retrasado debido a que el medio esta ocupado. La cuenta representada por una instancia de esta variable no incluye tramas implicadas en colisiones. Este contador no se incrementa cuando la interfaz esta operando en modo full-duplex.
dot3Stats LateCollisions	1.3.6.1.2.1.10.7.2.1.8	Counter32		Muestra el número de veces que una colisión es detectada en una interfaz en particular después que pase un slotTime en la transmisión de un paquete. Una colisión (posterior) incluida en una cuenta representada por una

				instancia de esta variable también es considerada como una colisión (genérica) para propósito de otras estadísticas asociadas con colisiones. Este contador no se incrementa cuando la interfaz esta operando en modo full-duplex.
dot3Stats Excessive Collisions	1.3.6.1.2.1.10.7.2.1.9	Counter32		Muestra una cuenta de tramas para las cuales la transmisión sobre una interfaz en particular falla debido a colisiones excesivas. Este contador no se incrementa cuando la interfaz esta operando en modo full-duplex. Las discontinuidades en el valor de este contador pueden ocurrir durante la reinicialización del sistema de gestión, y otras veces como es indicado en el valor de la variable ifCounterDiscontinuityTime.
dot3Stats InternalMac TransmitErrors	1.3.6.1.2.1.10.7.2.1.10	Counter32		Muestra una cuenta de tramas para las cuales la transmisión sobre una interfaz en particular falla debido a un error transmitido de forma interna en la subcapa MAC. Una trama solo es contada por una instancia de esta variable si no es contada por la correspondiente instancia de las variables dot3StatsLateCollisions,dot3StatsExcessiveCollisions o dot3StatsCarrierSenseErrors. El significado exacto de la cuenta representada por una instancia de esta variable es de implementación específica. En particular, una instancia de esta variable puede representar una cuenta de errores de transmisión en una interfaz en particular que no fue contada de otra manera.
dot3Stats CarrierSense Errors	1.3.6.1.2.1.10.7.2.1.11	Counter32		Indica el número de veces que la condición de detección de portadora se perdió o nunca se estableció cuando se intento transmitir una trama sobre una interfaz en particular. La cuenta representada por una instancia de esta variable se

				incrementa al menos una vez por cada intento de transmisión, aun si la condición de detección de portadora fluctúa durante un intento de transmisión. Este contador no se incrementa cuando la interfaz esta operando en modo full-duplex.
dot3Stats Frame TooLongs	1.3.6.1.2.1.10.7.2.1.13	Counter32		Muestra una cuenta de tramas recibidas en una interfaz en particular que exceden el máximo tamaño permitido de trama. La cuenta representada por una instancia de esta variable se incrementa cuando la trama TooLong status es retornada por el servicio MAC a el LLC (u otro usuario MAC). Las tramas recibidas para las cuales se obtienen múltiples condiciones de error, de acuerdo a las convenciones de IEEE 802.3 Layer Management, son contadas exclusivamente de acuerdo al estatus de error presentado al LLC (logical link control).
dot3Stats Internal MacReceive Errors	1.3.6.1.2.1.10.7.2.1.16	Counter32		Muestra una cuenta de tramas para las cuales la recepción en una interfaz en particular falla debido a un error de interno de subcapa MAC recibido. Una trama solo es contada por una instancia de esta variable si no es contada por la instancia correspondiente de cualquiera de las variables: dot3StatsFrameTooLongs, dot3StatsAlignmentErrors, dot3StatsFCSErrors. El significado preciso de la cuenta representada por una instancia de esta variable es de implementación específica. En particular, una instancia de esta variable puede representar una cuenta de errores recibidos en una interfaz en particular que de otra manera no son contados.

SNMP (1.3.6.1.2.1.11)

NOMBRE	OID	SINTAXIS	ENUMERACIÓN	DESCRIPCIÓN
SnmpInPkts	1.3.6.1.2.1.11.1	Counter32		Muestra el número total de mensajes entregados a una entidad SNMP desde el servicio de transporte.
SnmpOutPkts	1.3.6.1.2.1.11.2	Counter32		Muestra el número total de mensajes SNMP que fueron aprobados desde una entidad de protocolo SNMP hasta el servicio de transporte.

RMON (1.3.6.1.2.1.16)

NOMBRE	OID	SINTAXIS	ENUMERACIÓN	DESCRIPCIÓN
EtherStats DropEvents	1.3.6.1.2.1.16.1.1.1.3	Counter32		Muestra el número total de eventos en los cuales los paquetes fueron suprimidos por el sondeo debido a falta de recursos. Este número no es necesariamente el número de paquetes suprimidos; solo es el número de veces que esta condición ha sido detectada.
EtherStats Octets	1.3.6.1.2.1.16.1.1.1.4	Counter32		Muestra el número total de octetos de datos (incluyendo aquellos en paquetes malos) recibidos en la red (excluyendo los bits de entramado pero incluyendo octetos FCS). Esta variable puede ser usada como un estimado razonable de la utilización de Ethernet. Si se desea una precisión mayor, las variables etherStatsPkts y etherStatsOctets deben ser muestreadas antes y después de un intervalo común.
EtherStats Pkts	1.3.6.1.2.1.16.1.1.1.5	Counter32		Muestra el número total de paquetes (incluyendo los paquetes malos, paquetes de broadcast, y paquetes multicast) recibidos.
EtherStats BroadcastPkts	1.3.6.1.2.1.16.1.1.1.6	Counter32		Muestra el número total de paquetes válidos recibidos que fueron dirigidos a la dirección broadcast. No se incluyen los paquetes multicast.

EtherStats MulticastPkts	1.3.6.1.2.1.16.1.1.1.7	Counter32		Muestra el número total de paquetes validos recibidos que fueron dirigidos a una dirección multicast. Este número no incluye paquetes dirigidos a la dirección broadcast.
EtherStats CRC AlignErrors	1.3.6.1.2.1.16.1.1.1.8	Counter32		Muestra el número total de paquetes recibidos que tuvieron una longitud (excluyendo bits de entramado, pero incluyendo octetos FCS) entre 64 y 1518 octetos, pero tuvieron o una mala secuencia de chequeo de trama (FCS) con un número integral de octetos (FCS error) o una mala FCS con un número no integral de octetos (Alignment Error).
EtherStats UndersizePkts	1.3.6.1.2.1.16.1.1.1.9	Counter32		Muestra el número total de paquetes recibidos que eran de menos de 64 octetos de longitud (excluyendo bits de entramado, pero incluyendo octetos FCS) y estaban bien formados.
EtherStats OversizePkts	1.3.6.1.2.1.16.1.1.1.10	Counter32		Muestra el número total de paquetes recibidos que eran mayores a 1518 octetos (excluyendo bits de entramado, pero incluyendo octetos FCS) y estaban bien formados.
EtherStats Fragments	1.3.6.1.2.1.16.1.1.1.11	Counter32		Muestra el número total de paquetes recibidos que eran de menos de 64 octetos de longitud (excluyendo bits de entramado, pero incluyendo octetos FCS) y tenían o una más secuencias de chequeo de trama (FCS) con un número integral de octetos (error FCS) o una mala FCS con un número de octetos no integral (Alignment Error). Es enteramente normal para la variable etherStatsFragments el incrementarse. Esto se debe a que cuenta tanto runts(los cuales son acontecimientos normales debido a las colisiones) como golpes de ruido.

EtherStats Jabbers	1.3.6.1.2.1.16.1.1.1.12	Counter32		Muestra el número total de paquetes recibidos que eran de más de 1518 octetos de longitud (excluyendo bits de entramado, pero incluyendo octetos FCS) y tenían o una más secuencias de chequeo de trama (FCS) con un número integral de octetos (error FCS) o una mala FCS con un número de octetos no integral (Alignment Error). Observe que esta definición del jabber es diferente que la definición en IEEE-802.3 sección 8.2.1.5 (10BASE5) y sección 10.3.1.4 (10BASE2). Estos documentos definen el jabber como la condición donde cualquier paquete excede 20 ms. El rango permitido para detectar el jabber esta entre 20 ms y 150 ms.
EtherStats Collisions	1.3.6.1.2.1.16.1.1.1.13	Counter32		Indica el número total de colisiones mejor estimado (más preciso) en este segmento Ethernet. El valor retornado dependerá de la localización del sondeo RMON. Un sondeo localizado juega un rol mucho menor cuando se considera el 10BASE-T. 14.2.1.4 (10BASE-T) of IEEE standard 802.3 que define una colisión como la presencia simultanea de señales sobre los circuitos DO (data output) y RD (received data) (trasmisión y recepción en el mismo tiempo).
EtherStats Dropped Frames	1.3.6.1.2.1.16.1.4.1.1	Counter32		Muestra el número total de tramas que fueron recibidas por la sonda y por lo tanto no se contabilizó para entrar en *StatsDropEvents, pero para la cual la sonda escogió no contar para esta entrada por cualquier razón. Con frecuencia, este evento ocurre cuando la sonda no cuenta con algunos recursos y decide liberar la carga desde este grupo. Esta cuenta no incluye paquetes que no fueron contados debido a que tenían errores de capa MAC. A

				diferencia del contador dropEvent, este número, es el número exacto de tramas omitidas (dropped).
EtherHistory DropEvents	1.3.6.1.2.1.16.2.2.1.4	Counter32		Muestra el número total de eventos en los que los paquetes fueron suprimidos por la sonda debido a una falta de recursos durante este intervalo de muestreo. Este número no es necesariamente el número de paquetes suprimidos, solo es el número de veces que esta condición ha sido detectada.
EtherHistory Octets	1.3.6.1.2.1.16.2.2.1.5	Counter32		Muestra el número total de octetos de datos (incluyendo aquellos en paquetes defectuosos) recibidos en la red (excluyendo los bits de entramado pero incluyendo los octetos FCS).
EtherHistory Pkts	1.3.6.1.2.1.16.2.2.1.6	Counter32		Muestra el número de paquetes (incluyendo los defectuosos) recibidos durante este intervalo de muestreo.
EtherHistory Broadcast Pkts	1.3.6.1.2.1.16.2.2.1.7	Counter32		Muestra el número de paquetes buenos recibidos durante este intervalo de muestreo que fueron dirigidos a la dirección de broadcast
EtherHistory MulticastPkts	1.3.6.1.2.1.16.2.2.1.8	Counter32		Muestra el número de paquetes buenos recibidos durante este intervalo de muestreo que fueron dirigidas a una dirección multicast. Este número no incluye paquetes direccionados a la dirección de broadcast.
EtherHistory CRCAlign Errors	1.3.6.1.2.1.16.2.2.1.9	Counter32		Muestra el número de paquetes recibidos durante este intervalo de muestreo que tenían una longitud (excluyendo los bits de entramado pero incluyendo los octetos FCS) entre 64 y 1518 octetos, pero que tenían o una mala FCS con un número integral de octetos (error FCS) o una mala FCS con un número no integral de octetos (error de alineación).
EtherHistory UndersizePkts	1.3.6.1.2.1.16.2.2.1.10	Counter32		Muestra el número de paquetes recibidos durante

				este intervalo de muestreo que tenían menos de 64 octetos de longitud (excluyendo los bits de entramado pero incluyendo los octetos FCS) y estaban de lo contrario bien formados.
EtherHistory OversizePkts	1.3.6.1.2.1.16.2.2.1.11	Counter32		Muestra el número de paquetes recibidos durante este intervalo de muestreo que tenían más de 1518 octetos de longitud (excluyendo los bits de entramado pero incluyendo los octetos FCS) pero estaban de lo contrario bien formados.
EtherHistory Fragments	1.3.6.1.2.1.16.2.2.1.12	Counter32		Muestra el número de paquetes recibidos durante este intervalo de muestreo que tenían menos de 64 octetos de longitud (excluyendo los bits de entramado pero incluyendo los octetos FCS) que tenían una mala FCS con un número integral de octetos(error FCS) o una mala FCS con un número no integral de octetos (error de alineación). Es totalmente normal para la variable etherHistoryFragments el incrementarse. Esto se debe a que cuenta ambos enanos (runts) los cuales son acontecimientos normales debido a las colisiones y a los golpes de ruido.
EtherHistory Collisions	1.3.6.1.2.1.16.2.2.1.14	Counter32		Muestra la mejor estimación del número total de colisiones en este segmento Ethernet durante este intervalo de muestreo. El valor retornado dependerá de la ubicación de la sonda RMON. Así una sonda ubicada en un puerto de repetidor puede grabar mas colisiones que las que una sonda conectada a una estación en el mismo segmento podría. La ubicación de la sonda juega un papel mucho más pequeño cuando se considera los documentos que definen una colisión como la presencia simultanea de señales en los circuitos DO y RD (transmitiendo y recibiendo

				al mismo tiempo). Una estación 10BASE-T solo puede detectar colisiones cuando esta transmitiendo. Por lo tanto las sondas ubicadas en una estación y un repetidor, deben reportar el mismo número de colisiones.
EtherHistory Utilization	1.3.6.1.2.1.16.2.2.1.15	Integer32		Muestra el mejor estimado de la utilización media de la capa física en esta interfaz durante este intervalo de muestreo, en porcentaje.
HistoryControl Dropped Frames	1.3.6.1.2.1.16.2.5.1.1	Counter32		Muestra el número total de tramas que fueron recibidas por la sonda y por esa razón no fueron contadas para las StatsDropEvents, pero para las cuales la sonda eligió no contar para esta entrada por cualquier razón. Con frecuencia este evento ocurre cuando la sonda no cuenta con algunos recursos y decide deshacerse de la carga para esta colección. Esta cuenta no incluye paquetes que no fueron contados debido a que tenían errores de capa MAC. A diferencia del contador dropEvents counter, este número es el número exacto de tramas suprimidas.
AlarmRising Threshold	1.3.6.1.2.1.16.3.1.1.7	Integer32		Indica un umbral para la estadística muestreada. Cuando el valor muestreado actual es mayor o igual a este umbral, y el valor en el último intervalo de muestreo era menos que este umbral, un solo evento será generado. Un solo evento también será generado si la primera muestra después de que esta entrada llegue a ser válida es mayor o igual a este umbral y al alarmStartupAlarm asociado es igual a risingAlarm(1) o a risingOrFallingAlarm(3). Después de que se genere un evento de subida, otro evento así no será generado hasta que el valor muestreado caiga por debajo de este umbral y alcance el

				alarmFallingThreshold. Este objeto no puede ser modificado si el objeto asociado del alarmStatus es igual a valid (1).
AlarmFalling Threshold	1.3.6.1.2.1.16.3.1.1.8	Integer32		Indica un umbral para la estadística muestreada. Cuando el valor muestreado actual es mayor o igual a este umbral, y el valor en el último intervalo de muestreo era menos que este umbral, un solo evento será generado. Un solo evento también será generado si la primera muestra después de que esta entrada llegue a ser válida es mayor o igual a este umbral y al alarmStartupAlarm asociado es igual a risingAlarm(1) o a risingOrFallingAlarm(3). Después de que se genere un evento de subida, otro evento así no será generado hasta que el valor muestreado caiga por debajo de este umbral y alcance el alarmFallingThreshold. Esta variable no puede ser modificada si la variable asociado del alarmStatus es igual a valid (1).

RIP2 (1.3.6.1.2.1.23)

NOMBRE	OID	SINTAXIS	ENUMERACIÓN	DESCRIPCIÓN
rip2GlobalQueries	1.3.6.1.2.1.23.1.2	Counter32		Indica el número de respuestas enviadas a consultas RIP desde otros sistemas.

ANEXO B. Conceptos generales sobre sintaxis de variables MIB.

En este anexo se presentan los tipos de datos y sintaxis utilizadas para la definición y clasificación de las variables de la MIB.

1. ASN.1 (NOTACIÓN DE SINTAXIS ABSTRACTA 1)

Es conveniente realizar una referencia acerca del lenguaje de definición de notaciones como es el ASN.1. Se trata de un Lenguaje OSI para describir estándares y tipos de datos independientemente de las implementaciones, las estructuras de computadores específicos y técnicas de representación. Es descrito en el estándar Internacional de ISO 8824.

ASN.1 encierra más que definiciones de sintaxis, como lenguaje de programación fue desarrollado y estandarizado para conformar la estructura de estas bases de información.

El SNMP utiliza un subconjunto bien definido de dicho lenguaje, incluyendo un subconjunto más complejo para la descripción de objetos gestionados y para describir las unidades de datos de protocolo (PDU's) utilizadas para gestionar esos objetos.

Con el deseo de facilitar una futura transición a protocolos de gestión de redes basados en OSI, se procedió a la definición en el lenguaje ASN.1 de un SMI estándar de Internet y de una MIB.

2. MÓDULOS DE INFORMACIÓN

Existen tres clases de módulos ASN.1, también llamados ***Módulos de Información***, definidos por el SMI:

- *Módulos MIB*. que definen una colección de objetos de administración afines.

- *Sentencias de Conformidad*. que definen un conjunto de requisitos de los nodos con respecto a uno o más módulos MIB.

- *Sentencias de Capacidad*. que describen la capacidad de un nodo para implementar los objetos definidos en uno o más módulos MIB.

Por supuesto, estas funciones deberían estar combinadas en un sólo módulo.

3. TIPOS DE DATOS

Los tipos de datos utilizados en las definiciones de los OID son:

- **Integer32 e integer.** El tipo Integer32 representa información de valor entero entre -2^{31} y $2^{31}-1$ inclusive (-2147483648 a 2147483647 decimales). Este tipo es indistinguible del tipo INTEGER.

El tipo INTEGER también puede usarse para representar información de valor entero, si contiene enumeraciones numéricas, o si se crea un subtipo para restringir más que el tipo Integer32. En el caso anterior, sólo esas enumeraciones de cantidades pueden estar presentes como un valor. Aunque se recomienda que los valores enumerados empiecen en 1 y estén numerados de forma consecutiva, cualquier valor válido para Integer32 es permitido como valor enumerado y, además, los valores enumerados no necesitan ser asignados de forma consecutiva.

Finalmente, el carácter guión no se permite como una parte del nombre de ninguna enumeración de cantidades.

- **Octet string.** El tipo OCTET STRING representa datos binarios o textuales arbitrarios. Aunque SMI no especifica ninguna limitación del tamaño para este tipo, los diseñadores de

MIB deben comprender que puede haber limitaciones en la implementación e interoperabilidad para tamaños superiores a 255 octetos.

- **Object identifier.** El tipo OBJECT IDENTIFIER representa administrativamente a los nombres asignados. Cualquier instancia de este tipo puede tener, como mucho, 128 sub-identificadores. Además, cada sub-identificador no debe exceder el valor 2^{32}-1 (4294967295 decimal).

- **Bit string.** El tipo BIT STRING representa una enumeración de bits. Esta colección contendrá valores no negativos, consecutivos y comenzando por el cero. Sólo esos bits enumerados pueden estar presentes en un valor. Un requisito en los módulos MIB "estándar" es que el carácter guión no está permitido como parte del nombre de una enumeración de bits.

- **Ip address.** El tipo IP Address representa una dirección Internet de 32-bits. Esta es representada como un OCTET STRING de longitud 4, en formato de red. El tipo IP Address es un tipo etiquetado por razones históricas. Deben representarse las direcciones de red usando una llamada a la macro TEXTUAL-CONVENTION.

- **Counter32.** El tipo Counter32 representa un entero no negativo que se incrementa de forma monótona hasta el valor máximo de 2^{32}-1 (4294967295 decimal), cuando se alcanza dicha cifra, se vuelve al cero y se incrementa de nuevo.

Los contadores no tienen definido un valor "inicial", y así, un valor solitario de un Contador no representa (en general) ningún volumen de información.

Suelen ocurrir discontinuidades en el incremento monótono del valor en la reinicialización del sistema de gestión, y en otras ocasiones como se especificó en la descripción de un tipo de objeto que usa este tipo **ASN.1**.

Si los otros casos pueden ocurrir, por ejemplo, la creación de una instancia de objeto en un momento distinto al de la reinicialización, entonces, el objeto correspondiente debe definirse con un valor de cláusula SINTAX de TimeStamp indicando el tiempo de la última discontinuidad.

El valor de la cláusula MAX-ACCESS para los objetos con un valor de Counter32 de su cláusula SINTAX es siempre de sólo lectura. La cláusula DEFVAL no está permitida para los objetos con un valor de su cláusula SINTAX de Counter32.

- **Gauge32.** El tipo Gauge32 representa un entero no negativo que puede aumentar o disminuir, pero nunca excederá un valor máximo. El valor máximo no puede superar 232-1 (4294967295 decimal).

El máximo valor de un Gauge será siempre el valor máximo que la información que contiene pueda alcanzar; si la información contenida disminuye por debajo del máximo valor, el Gauge también disminuye.

- **Timeticks.** El tipo TimeTicks representa un entero no negativo que representa el tiempo, modulo 232 (4294967296 decimal), en centésimas de un segundo, entre dos épocas. Cuando se definen objetos que usan este tipo, la descripción del objeto identifica las dos épocas a las que se hace referencia.

Por ejemplo, se define el convenio textual de TimeStamp qué está basado en el tipo TimeTicks. Con un TimeStamp, la primera referencia a una época se define por Ej. Cuando sysUpTime del MIB-II era cero, y la segunda referencia a una época se define como el valor actual de SysUptime.

- **Counter64.** El tipo Counter64 representa un entero no negativo que aumenta de forma monótona hasta alcanzar el valor máximo de 2^{64}-1 (18446744073709551615 decimal),

cuando supera ese valor, se comienza de nuevo desde cero. Los contadores no tienen definido un valor inicial, y así, un solo valor de un contador no tiene, en general, ningún valor de información.

Discontinuidades en la monotonía creciente del valor del contador, son provocadas normalmente por la reinicialización del sistema de gestión, y también al especificar la descripción de un tipo de objeto que usa este tipo de dato. Si alguna de estas situaciones ocurre, por ejemplo, la creación de una instancia de un objeto en un momento distinto al de la reinicialización, entonces, el objeto correspondiente debe definirse con un valor de cláusula SINTAX de TimeStamp indicando el tiempo de la última discontinuidad

El valor de la cláusula de MAX-ACCESS para los objetos con un valor de Counter64 en su cláusula SINTAX siempre es de sólo lectura ("read-only").

Un requisito en los módulos MIB estándar es que el tipo Counter64 sólo puede usarse si la información a contener diera la vuelta en menos de una hora usando el tipo Counter32. La cláusula DEFVAL no está permitida para objetos con un valor de Counter64 en su cláusula SINTAX.

- **Uinteger32.** El tipo UInteger32 representa información de valor entero entre 0 y 232-1 inclusive (0 a 4294967295 decimales). Se trata de un valor entero sin signo.

- **Max-access.** La cláusula MAX-ACCESS, que debe estar presente, define si tiene privilegios para leer, escribir y/o crear una instancia del objeto. Éste es el máximo nivel de acceso para el objeto. (Este nivel máximo de acceso es independiente de cualquier política administrativa de accesos.)

El valor "**read-write**" (lectura-escritura) indica que está permitida la lectura y la escritura, pero no la creación. Un valor "**read-create**" (lectura-creación) indica posibilidad de lectura,

escritura y creación. Un valor de "**not-accessible**" indica o un objeto auxiliar o un objeto que sólo es accesible por una notificación.

Estos valores están ordenados de menor a mayor: "not-accessible", "read-only", "read-write", "read-create".

Si algún objeto en una fila conceptual tiene "**read-create**" como su nivel máximo de acceso, entonces ningún otro objeto de la misma fila conceptual puede tener un nivel máximo de acceso de "read-write". ("read-create" está por encima de "read-write")

- **Convenciones textuales.** Aunque los tipos de datos específicos de la aplicación, contenidos en el SMI son usados, la experiencia en la creación de módulos MIB muestra que, a veces, es conveniente definir tipos de datos con sintaxis similar a la estándar, pero con una semántica mucho más precisa. Estos tipos, se denominan *convenciones textuales*.

Su codificación es idéntica a la de los otros tipos, pero dentro del MIB, poseen una semántica especial, la cual es capturada por la macro *TEXTUAL-CONVENTION*.

Ejemplo:
Display String: = TEXTUAL-CONVENTION
DISPLAY-HINT "255a"
STATUS current
DESCRIPTION
"Representa información textual, y ningún objeto puede superar los 255 caracteres de longitud"
SYNTAX OCTET STRING (SIZE (0.255))

4. IDENTIFICACIÓN DE INSTANCIAS DE OBJETOS

Para conocer el valor de una instancia de un objeto, se necesita identificar la instancia, usando el ***OBJECT IDENTIFIER***. (OID)

- **CONVENCIONES PARA IDENTIFICAR INSTANCIAS.**

- **Objetos Escalares (o Variables Simples).** tienen sólo una instancia asociada con cada objeto escalar, que se identifica por concatenar un valor 0 al *OBJECT IDENTIFIER*.

- **Objetos Columnares (o Tablas).** las instancias de estos objetos se identifican en una tabla por la cláusula INDEX, que se refiere a una fila en una tabla.

- **Tablas y Filas Conceptuales.** no tienen identificadores de instancias asociados.

- **Orden Lexicográfico.** los *OBJECT IDENTIFIERs* están ordenados en forma creciente en las MIBs SNMP.

5. MANIPULACIÓN DE TABLAS

En las tablas de la MIB-II, cada objeto está representado por una columna, y el valor de cada instancia de objeto por una fila. Se atraviesa completamente cada columna a lo largo, y luego uno se mueve a la columna siguiente.

Para agregar un valor a una instancia, se ingresa el valor en una fila con una operación Set. Para borrar una entrada, nuevamente usando la operación Set, se pone el valor como inválido (se recomienda removerlo después).

6. DETALLES Y OBJETOS DE LA MIB-II (RFC 1213)

Cuando se definen nuevas MIBs, es necesario seguir algunas reglas, para permitir la coexistencia de múltiples versiones de MIBs:

- Los tipos de objetos viejos no se borran, pero deben ser removidos en las versiones siguientes.

- Las semánticas de los tipos de objetos viejos no deberían cambiar entre versiones. Sin embargo, si se necesita cambiar la semántica, deben formarse nuevos tipos de objetos.

En la MIB, sólo se definen los objetos esenciales, siguiendo los lineamientos provistos por la SMI para definir nuevos objetos.

Estos nuevos objetos pueden agregarse bajo el subárbol *{experimental 3}* o bajo *{enterprises 4.1}*.

A la MIB-II se le han agregado los tipos de datos *DisplayString* (*OCTET STRING* de caracteres ASCII imprimibles, de 0 a 255 octetos), y *PhyAddress* (*OCTET STRING* usado para representar direcciones físicas).

Los objetos en Internet se clasifican en diferentes grupos, bajo *{mgmt 2}*. Los objetos bajo estos grupos deben implementarse como un grupo. Ej.: si se implementa el grupo *TCP*, entonces todos los objetos bajo el grupo *TCP*, tales como *tcpRtoAlgorithm* y *tcpRtoMin*, deben ser implementados.

ANEXO C. Características de la monitorización de redes con SNMP.

1. DEFINICIÓN DE MONITORES DE TRÁFICO

Un Monitor es un instrumento que entrega datos de algún tipo (numérico, visual, auditivo) de un proceso o fenómeno. Un Monitor de Tráfico en Redes de Computadores es un instrumento que entrega datos acerca de la red en la cual está conectado. Estos datos pueden ser entregados en diversas formas, dependiendo del fin con el cual el monitor fue diseñado.

2. CLASIFICACIÓN DE MONITORES

Los monitores de Red se pueden clasificar según los criterios de Objetivo, Reporte, Intrusividad, Operación y Protocolos que miden.

- **Objetivo.**

- ***Monitor de Estados (variables).*** El objetivo es el de "avisar" situaciones de emergencia, como "caídas" de equipos o el sobrepaso de un umbral de alguna variable fijada por el administrador. Por ejemplo: La NO-RESPUESTA de un Router o la superación del umbral de 40% en el porcentaje de colisiones.

Para ello se ocupa generalmente el estándar SNMP (Simple Network Management Protocol) que, está orientado a monitorizar y configurar el equipamiento físico de una red (bridges, Routers, hubs y estaciones de trabajo).

Dentro de los Monitores SNMP Comerciales se destacan HP OpenView, SUNNET Manager y Cisco Works. Cabe mencionar que el SNMP no sólo sirve para monitorear y/o configurar variables de los equipamientos de la red física sino que también para mostrar el

tráfico histórico en forma gráfica acerca de lo que se ha transportado por los equipos de la red, como routers, switches, etc.

- ***Monitor de Tráfico.*** El objetivo es el de registrar el tráfico de las redes, ya sea con fines estadísticos o de detección de congestión. Este tipo de monitor también puede tener "alarmas" y por ende, enviar una señal al administrador cuando una variable monitoreada haya excedido un umbral prefijado (considerado como alarmante).

- **Reporte.**

- ***Histórico.*** Este tipo de monitor entrega informes o resúmenes de la actividad histórica de la red, ya sea por: hora, día, semana o mes. La actividad mencionada puede corresponder a tráfico, alarmas ocurridas, etc.

El reporte generalmente consiste en archivos de tipo texto, dado que éste puede ser ingresado a otro programa para transformarlo a gráficos. Los últimos monitores de este tipo entregan sus reportes en formato HTML y generan automáticamente los gráficos usando programas propios o del sistema. Un monitor histórico debe entregar la mayor cantidad de información posible, ya sea en forma de texto o de gráfico, ya que la información entregada puede ser ingresada a una base de datos.

- ***Tiempo real.*** Un monitor de este tipo entrega datos de muy reciente ocurrencia, desde 1 segundo hasta 10 minutos, con el objeto de detectar y corregir los problemas cuanto antes. En algunos casos, como en las "alarmas" de umbrales, las variables son de tiempo corto y por lo tanto son inmediatamente avisadas.

En otros casos, si la variable tiene un tiempo de "ejecución", como la transmisión de un paquete, se debe esperar la transmisión completa antes de poder tomar acciones. Por estas razones, un monitor de tiempo real no sólo abarca a los monitores "instantáneos", sino también a los que deben monitorear variables que requieren de un tiempo determinado de

ejecución.

- **Intrusividad**

- ***Intrusivo.*** Es aquel monitor que interviene en el proceso o fenómeno a monitorear, vale decir, actúa como agente activo. En este caso, el monitor afecta la medición, haciéndola no confiable o no representativa del proceso. Por ejemplo, un monitor de tráfico local se considera intrusivo si ocupa la red para hacer mediciones.

Sin embargo, si un monitor ocupa la red para obtener datos, no significa que su medición sea intrusiva, ya que esto depende del proceso que desee medir. Por ejemplo: si el monitor debe medir el tráfico que es cursado por un router y consulta a éste último por esa información (usando la red local), su medición no es intrusiva.

- ***No Intrusivo.*** Un monitor es no intrusivo si no interviene el proceso o fenómeno a monitorear, vale decir, si actúa como agente pasivo. Por ejemplo: un monitor de tráfico local no es intrusivo si no utiliza la red para medir su tráfico local.

Si el monitor es un monitor del estado de un proceso en UNIX, el monitor en cuestión no debe interactuar con tal proceso, porque afectaría su ejecución normal.

- **Operación.**

Esta clasificación se refiere a la operación del monitor, en cuanto a dónde despliega sus datos y dónde es configurado por un administrador.

- ***Local.*** Si la operación del monitor es local, entonces significa que muestra los datos en el mismo lugar de donde los obtiene.

En el caso de un monitor de tráfico, esto significa que los datos se despliegan en el mismo computador que obtiene los datos.

- ***Remota.*** La operación de este tipo de monitores se realiza en forma remota. En el caso de un monitor de tráfico, esto significa que los datos son desplegados en otro computador distinto al que obtiene los datos.

Lo anterior requiere ocupar la red para comunicar estos 2 computadores, por lo que dependiendo del tipo de medición, será también clasificado como intrusivo o no intrusivo.

- **Protocolos.**

Esta clasificación permite destacar los tipos de protocolos de redes con que el monitor puede trabajar. La clasificación contempla mencionar la compatibilidad con los siguientes protocolos de red: Las 4 versiones de Ethernet; Versión II, Novell RAW, IEEE 802.3 y SNAP. TCP/IP; estándar para aplicaciones Internet (Intranet).IPX; estándar para redes NovellNETbeui (NETbios); estándar para redes Microsoft.

3. Monitores Compatibles con SNMP (Simple Network Management Protocol).

El SNMP es un protocolo ampliamente conocido y muy utilizado en ambientes donde la red es muy grande (30 hubs, 20 Routers, 20 Servidores, etc.) y además donde la estabilidad física de la Red es vital (bancos, edificios corporativos, grandes empresas).

El SNMP consta de 3 elementos: los "Agentes", el "Manager o gestor" y las MIB (Management Information Base). Los Agentes son programas que son instalados en Routers, Hubs, Estaciones de Trabajo, etc., que permiten una interfaz entre el protocolo SNMP y la configuración local del equipo.

El "gestor" es el programa central que consulta (o configura) las variables de cada nodo, interactuando con el "Agente" respectivo. La MIB es el "árbol" de especificaciones de la red, donde la raíz del árbol contiene las variables más globales de lo que sucede en la red y

las "hojas" del árbol corresponden a la información detallada de cada nodo en la red (los Agentes). La MIB es conceptual y es mostrada por el gestor y éste debe reunir la información de los Agentes.

Existen cinco tipos de paquetes SNMP llamados PDU (Protocol Data Unit): dos para leer datos de los nodos, dos para configurar datos de los nodos y uno para los TRAP (umbrales prefijados que activan una señal de alarma), que usa el Agente para enviar eventos (programados por el administrador) como la "inicialización de un nodo" o la falla de una estación, etc.

Así, si el administrador desea saber el valor de una variable de un Router, usa SNMP para enviar un PDU a ese nodo. El agente del Router busca en su MIB el valor actual de la variable y envía el valor en otro PDU al gestor

Lo que se puede hacer en cada nodo depende del agente respectivo, vale decir, que se puede programar enteramente un Router vía SNMP, consultar sus estadísticas de tráfico, se puede administrar un nodo UNIX (crear cuentas, verificar espacio en disco, etc.) vía SNMP sólo "leyendo" y "escribiendo" variables. Las ventajas del SNMP son su fácil uso y el hecho de que permite manejar los nodos físicos de la red en forma centralizada (remota).

Los gestores que se destacan son en su mayoría versiones comerciales, como HP Open View, SUNNET Manager, CISCO Works, IBM NetView. También existen otros de dominio público, como XSNMP, AARNet Traffic Monitoring para UNIX; NetGuardian y SNMPMan para Windows.

Podemos clasificar a los monitores SNMP según:

- *Objetivo*. Principalmente Monitor de Estados (o variables), ya que no están orientados a ser monitores de tráfico.

- *Reporte*. Histórico en su mayoría. En el caso de las alarmas o TRAP son de "Tiempo Real".

- *Intrusividad*. Se consideran Intrusivos, ya que utilizan bastante la red para obtener las variables. Se ha demostrado que la eficiencia (throughput) de la red baja cuando se utiliza SNMP.

- *Operación*. Claramente Remota.

- *Protocolos*. El SNMP es principalmente una aplicación TCP/IP, por lo que la gran mayoría de los gestores y Agentes de la red son TCP/IP. En general, cubre todos los tipos de redes Microsoft, TCP/IP y Novell.

En el caso de las versiones Ethernet, tanto el Manager como el Agente son construidos de acuerdo al dispositivo a manejar.

Las ventajas y desventajas que se presentan a continuación son establecidas con respecto a los otros monitores explicados.

- **Ventajas.**

- Utilizan un protocolo estándar que permite integrar varios Agentes de distintas marcas al gestor, compatibilizando el manejo centralizado de las redes.

- La plataforma de trabajo para el Administrador de la Red puede ser una estación UNIX o un simple PC con Windows 95.

- Constituye una plataforma de trabajo muy simple para la administración, donde entrega parámetros globales y específicos según la necesidad del administrador.

- **Desventajas.**

- Es claramente un monitor de estados, por lo que sólo puede medir variables globales de tráfico según la información entregada por el Agente respectivo, el cual no está destinado a medir en tiempo real ninguna clase de tráfico.

- La mayor cantidad de reportes que entrega son históricos, por lo que no sirve para monitorear transiciones y/o fluctuaciones del tráfico mientras ocurren. Sólo es posible enterarse después.

- Dado que los Agentes y las MIBS vienen del fabricante y el gestor sólo agrupa y despliega la información entregada, el SNMP no es una plataforma para poder medir variables de Protocolos y estándares en desarrollo o experimentación. El investigador debe programar su propio Agente y su propia MIB.

ANEXO D. Especificaciones de las Herramientas Software.

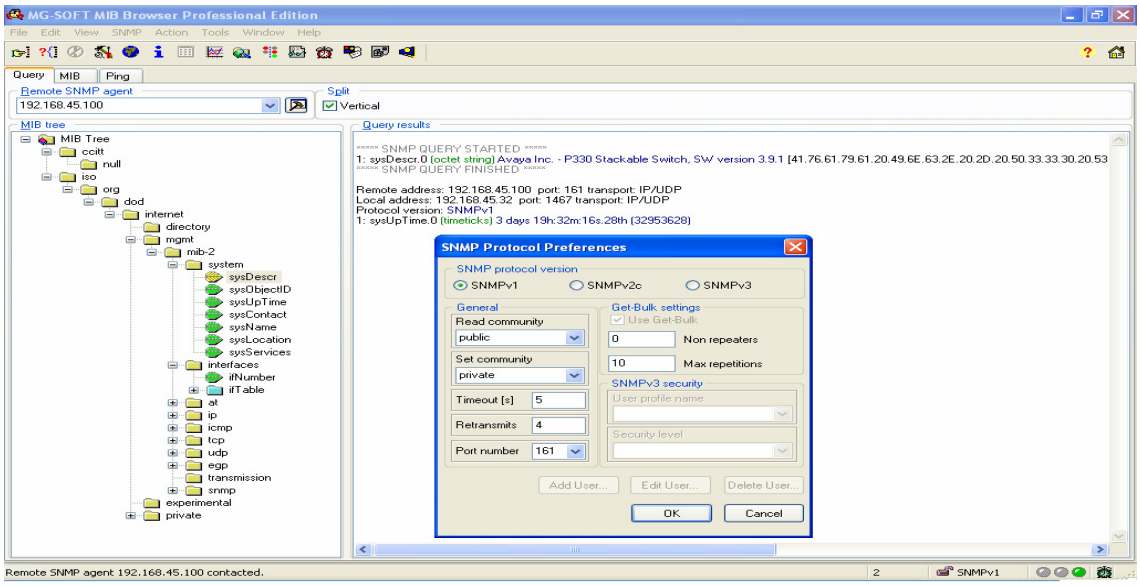
En este anexo se presentan las características de las herramientas software utilizadas en el desarrollo de este trabajo.

1. HERRRAMIENTAS SOFTWARE PARA LA MONITORIZACIÓN DE LAS VARIABLES MIB

1.1 MG SOFT

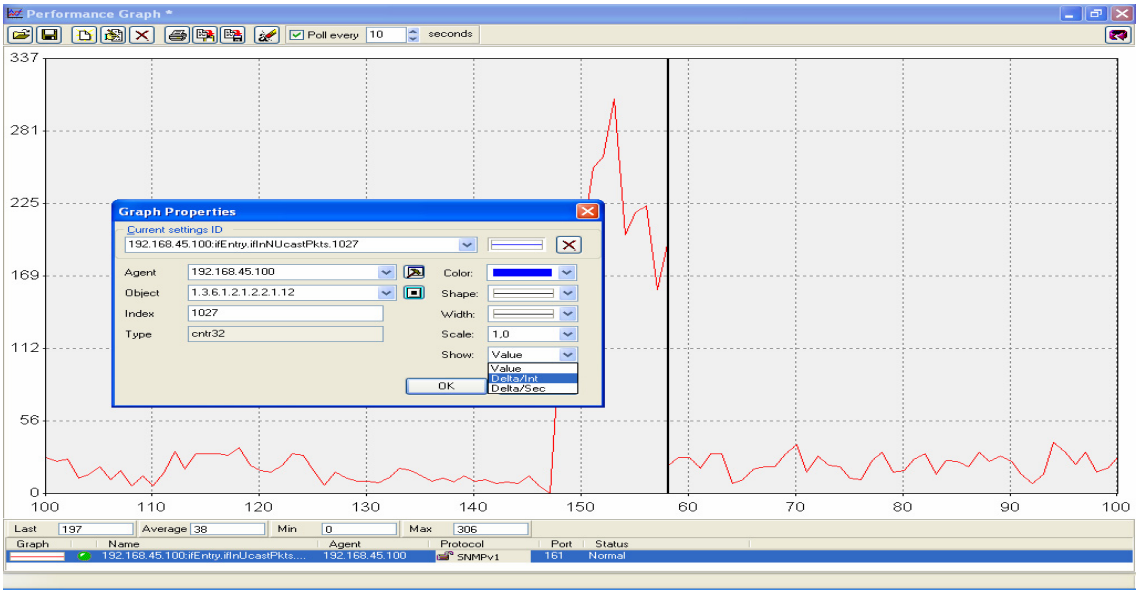
Este paquete cuenta con varias herramientas que serán explicadas a continuación.

1.1.1 MG-SOFT MIB Browser. Es un navegador de la MIB que corre sobre sistemas operativos Microsoft Windows (Windows 95, 98, NT, ME, 2000, XP y Windows Server 2003); y sobre sistemas operativos Linux (RedHat, Mandrake, SuSE, etc.). Permite establecer las operaciones SNMP Get, SNMP GetNext, SNMP GetBulk y SNMP Set. Además puede capturar SNMP Traps.

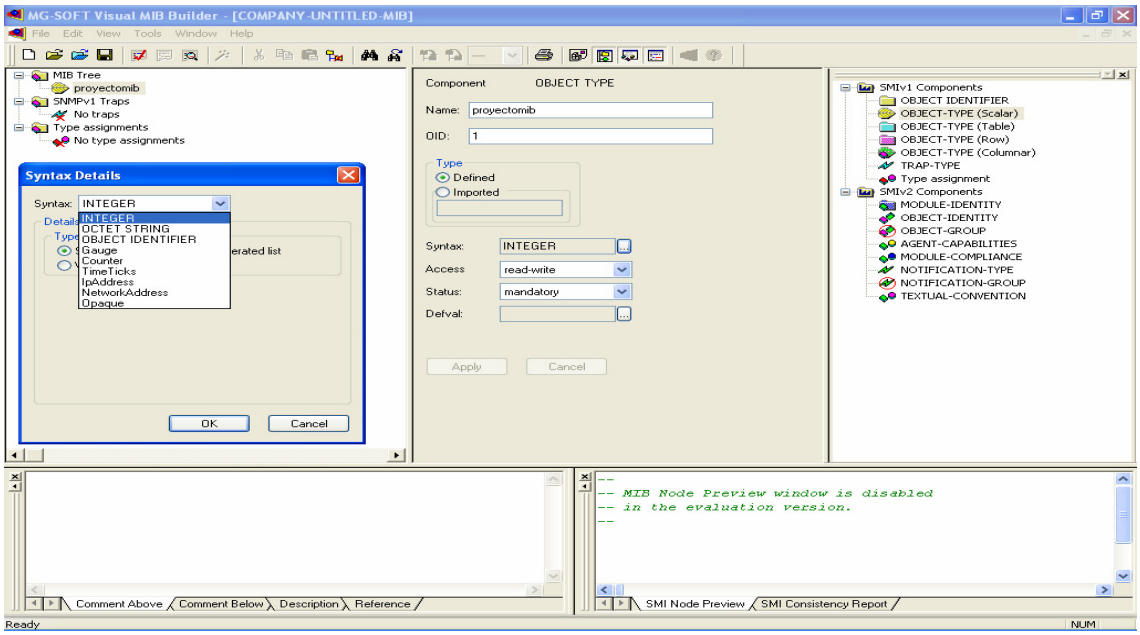


Entre sus capacidades esta la de monitorizar varios dispositivos mediante SNMP, y permite ver simultáneamente arreglos de tablas, tiene capacidad de guardar

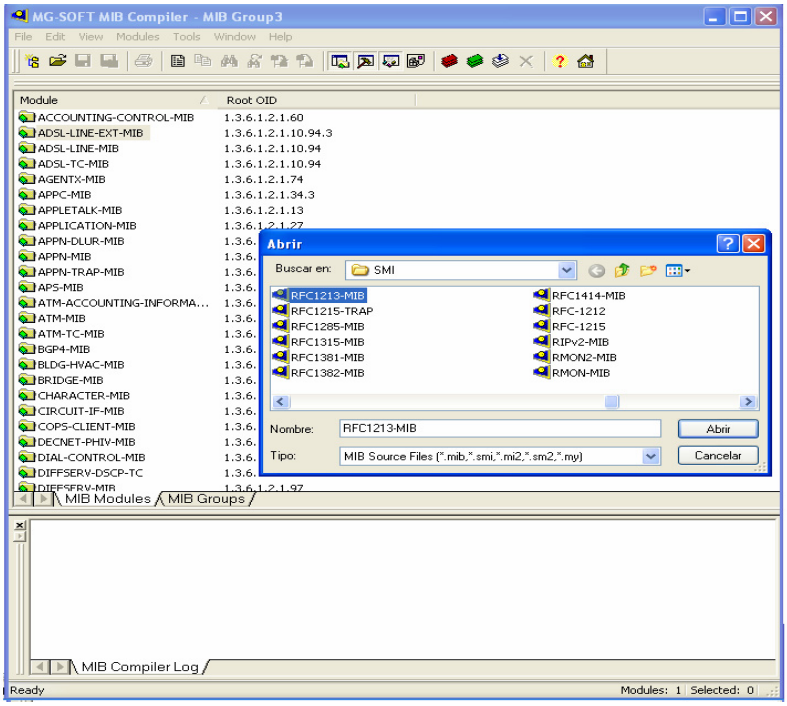
información y de presentación gráfica en tiempo real de valores numéricos encuestados, además permite la exploración para MIBs puesta en ejecución en los agentes.



1.1.2 MG-SOFT MIB Builder. Es otra herramienta complementaria que ofrece la posibilidad de configurar un SMI y adecuarlo para una red particular; también permite crear una rama propia (módulo), que con la ayuda de la herramienta (**MG-SOFT MIB Compiler**) se puede cargar al MG-SOFT MIB Browser en donde se puede ejecutar y correr con los demás módulos que se tengan activados en el dispositivo**.**



1.1.3 MG-SOFT MIB Compiler. Esta herramienta permite compilar cualquier archivo MIB de un fabricante especifico; este archivo debe ser cargado para ser utilizado por el MIB browser.



1.2 SOLARWINDS PROFESSIONAL EDITION 5.2

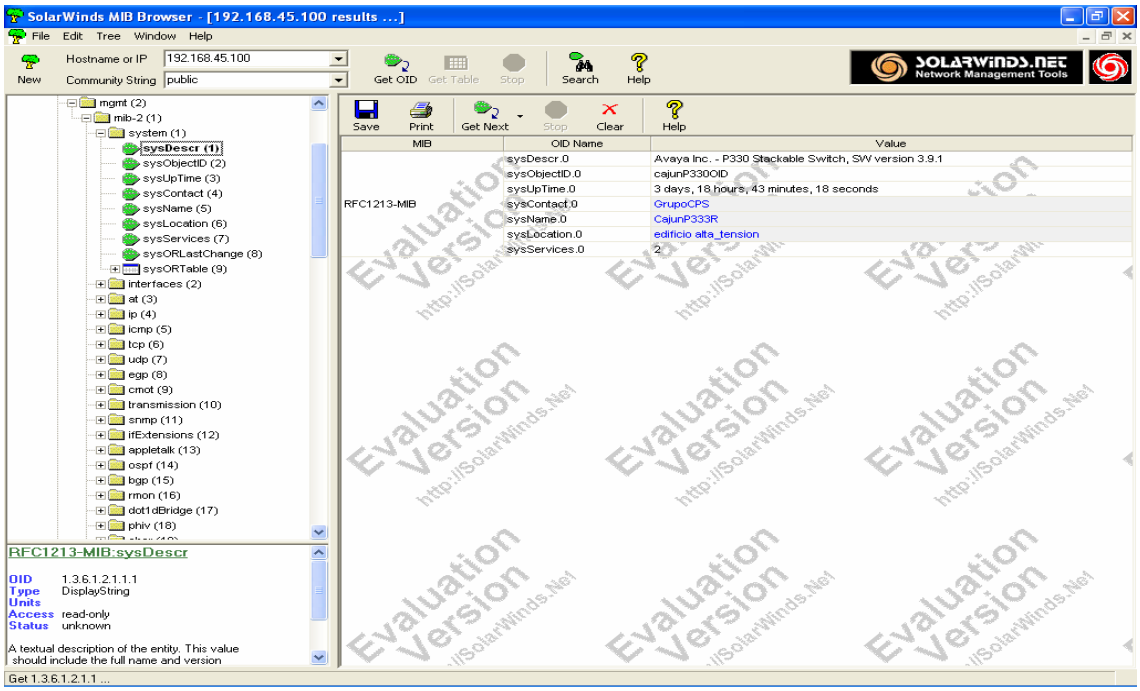
SolarWinds.Net desarrolla y ofrece un arsenal de herramientas de administración de red, de monitorización y de herramientas de descubrimiento de la red, para responder a los requisitos diversos de la ingeniería de hoy en las redes.

SolarWinds corre bajo sistemas operativos Windows 95 / 98 / NT / 2000 / Millennium/XP.

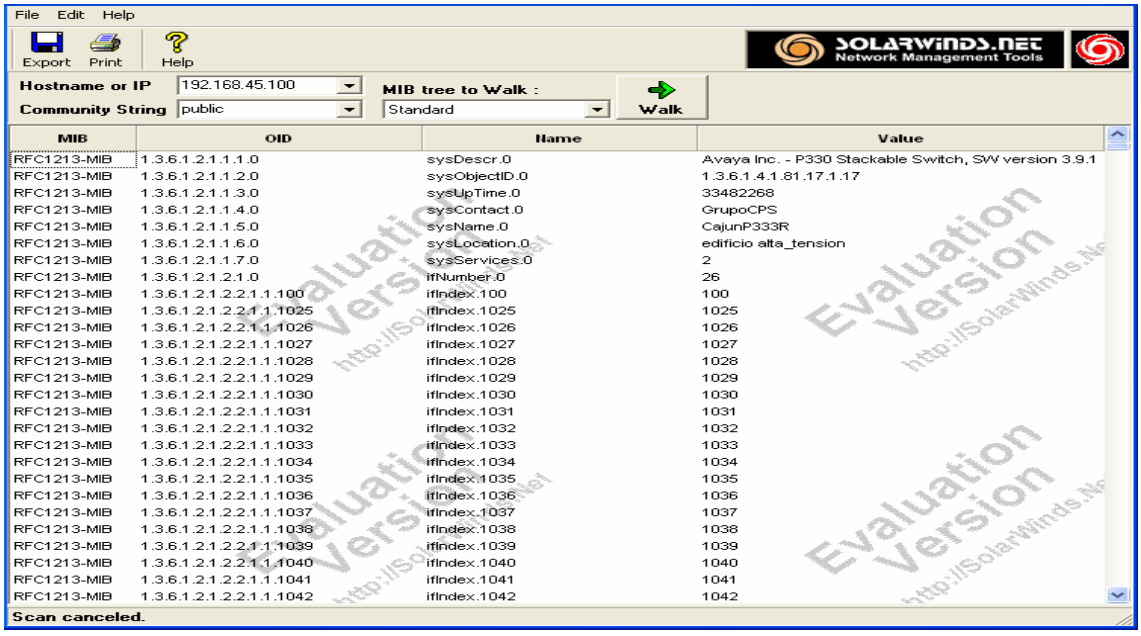
Entre sus 9 paquetes de herramientas, aquellos que brindan información sobre las variables de la MIB y que pueden servir para su monitorización se encuentran: MIB BROWSER y Performance Monitoring.

1.2.1 MIB BROWSER. Consta de las siguientes herramientas.

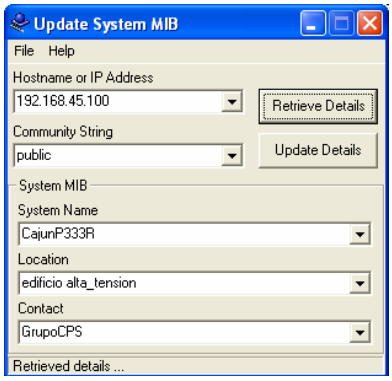
- **MIB browser.** Es un navegador completo de la MIB. Permite explorar el árbol y encuestar la información del OID, puede modificar remotamente valores del SNMP. La base de datos de la MIB de SolarWinds contiene más de 1.000 módulos MIB y más de 100.000 OIDs públicos y privados (de fabricante).



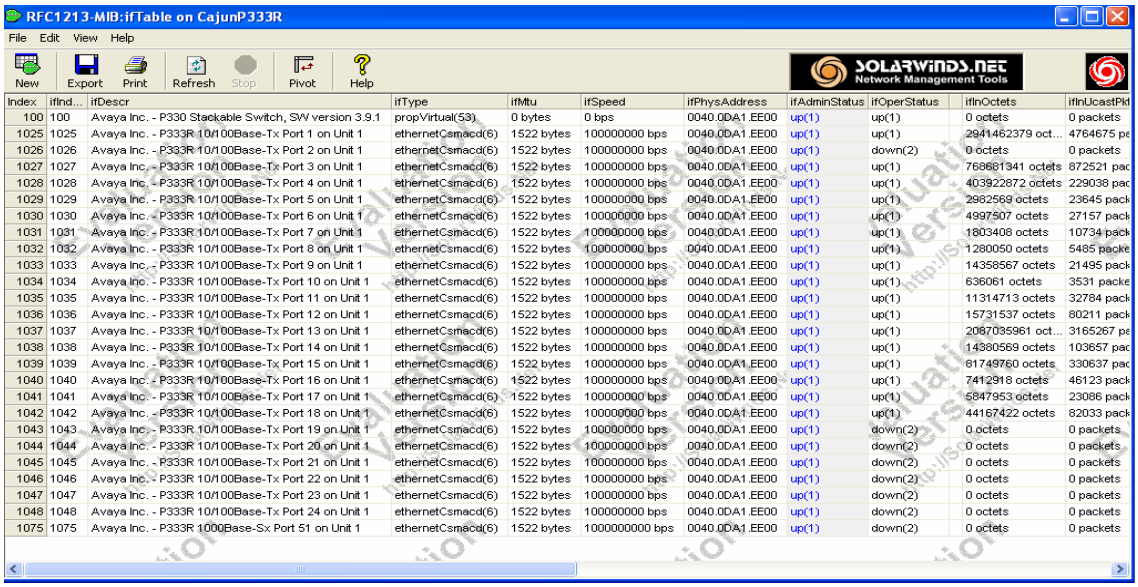
- **MIB walk.** Recorre el árbol del SNMP utilizando la base de datos de la MIB de Solarwinds y genera una tabla de todas las MIB y OID's soportadas en un dispositivo específico.



- **Update system MIB.** Esta herramienta se utiliza para fijar el nombre, el contacto, y la localización para los dispositivos con SNMP; impresoras de la red, los hubs, y los servidores terminales, etc.



- **MIB viewer.** Los valores de las variables de la MIB que conforman una tabla normalmente no se pueden visualizar como tal, por lo que esta herramienta los muestra de forma ordenada por sus respectivas filas dadas por el índice de la interfase y de columnas dadas por el OID de la variable. Tiene la misma información que el MIB browser, pero es un método más rápido.

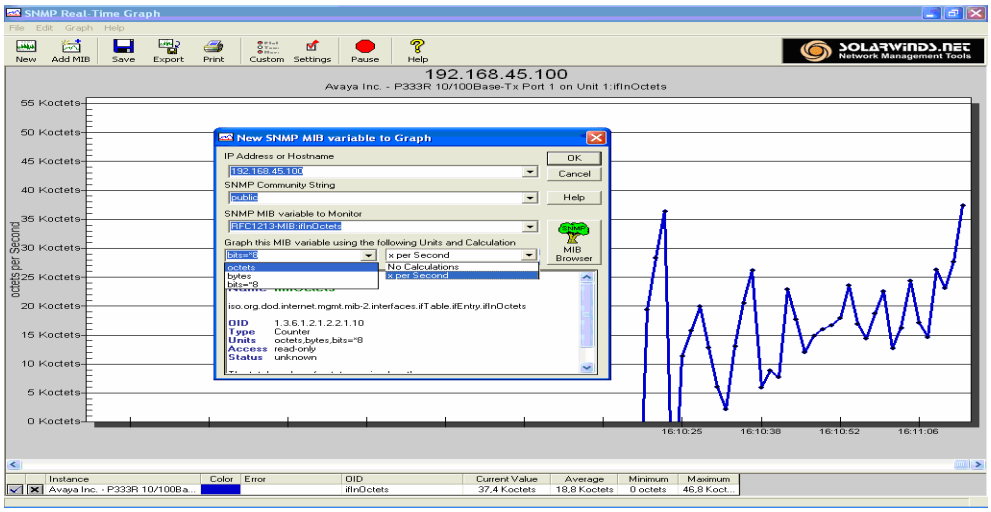


Index	ifInd...	ifDescr	ifType	ifMtu	ifSpeed	ifPhysAddress	ifAdminStatus	ifOperStatus	ifInOctets	ifInUcastPkt
100	100	Avaya Inc. - P330 Stackable Switch, SW version 3.9.1	propVirtual(53)	0 bytes	0 bps	0040.0DA1.EE00	up(1)	up(1)	0 octets	0 packets
1025	1025	Avaya Inc. - P333R 10/100Base-Tx Port 1 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	2941462379 oct...	4764675 pa
1026	1026	Avaya Inc. - P333R 10/100Base-Tx Port 2 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	down(2)	0 octets	0 packets
1027	1027	Avaya Inc. - P333R 10/100Base-Tx Port 3 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	768681341 octets	872521 pac
1028	1028	Avaya Inc. - P333R 10/100Base-Tx Port 4 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	403922872 octets	229038 pac
1029	1029	Avaya Inc. - P333R 10/100Base-Tx Port 5 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	2982569 octets	23645 pack
1030	1030	Avaya Inc. - P333R 10/100Base-Tx Port 6 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	4997507 octets	27157 pack
1031	1031	Avaya Inc. - P333R 10/100Base-Tx Port 7 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	1803408 octets	10734 pack
1032	1032	Avaya Inc. - P333R 10/100Base-Tx Port 8 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	1280050 octets	5485 packe
1033	1033	Avaya Inc. - P333R 10/100Base-Tx Port 9 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	14358567 octets	21495 pack
1034	1034	Avaya Inc. - P333R 10/100Base-Tx Port 10 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	636061 octets	3531 packe
1035	1035	Avaya Inc. - P333R 10/100Base-Tx Port 11 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	11314713 octets	32784 pack
1036	1036	Avaya Inc. - P333R 10/100Base-Tx Port 12 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	15731537 octets	80211 pack
1037	1037	Avaya Inc. - P333R 10/100Base-Tx Port 13 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	2067035961 oct...	3165267 pa
1038	1038	Avaya Inc. - P333R 10/100Base-Tx Port 14 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	14380569 octets	103657 pac
1039	1039	Avaya Inc. - P333R 10/100Base-Tx Port 15 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	81749760 octets	330637 pac
1040	1040	Avaya Inc. - P333R 10/100Base-Tx Port 16 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	7412918 octets	46123 pack
1041	1041	Avaya Inc. - P333R 10/100Base-Tx Port 17 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	5847953 octets	23086 pack
1042	1042	Avaya Inc. - P333R 10/100Base-Tx Port 18 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	up(1)	44167422 octets	82033 pack
1043	1043	Avaya Inc. - P333R 10/100Base-Tx Port 19 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	down(2)	0 octets	0 packets
1044	1044	Avaya Inc. - P333R 10/100Base-Tx Port 20 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	down(2)	0 octets	0 packets
1045	1045	Avaya Inc. - P333R 10/100Base-Tx Port 21 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	down(2)	0 octets	0 packets
1046	1046	Avaya Inc. - P333R 10/100Base-Tx Port 22 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	down(2)	0 octets	0 packets
1047	1047	Avaya Inc. - P333R 10/100Base-Tx Port 23 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	down(2)	0 octets	0 packets
1048	1048	Avaya Inc. - P333R 10/100Base-Tx Port 24 on Unit 1	ethernetCsmacd(6)	1522 bytes	100000000 bps	0040.0DA1.EE00	up(1)	down(2)	0 octets	0 packets
1075	1075	Avaya Inc. - P333R 1000Base-Sx Port 51 on Unit 1	ethernetCsmacd(6)	1522 bytes	1000000000 bps	0040.0DA1.EE00	up(1)	down(2)	0 octets	0 packets

1.2.2 PERFORMANCE MONITORING. Este paquete consta de un conjunto de herramientas que permiten monitorizar el desempeño de dispositivos mediante cálculos basados en información estadística suministrada por las variables de la MIB; en la mayoría de herramientas dentro de este paquete esta información es transparente para el usuario y por eso solo se tendrá en cuenta la herramienta SNMP GRAPH.

- **SNMP GRAPH.** El SNMP-Graph de SolarWinds es una herramienta de monitorización de datos en tiempo real, capaz de mostrar gráficamente datos de cualquier MIB (Management Information Base) simplemente seleccionando el dispositivo y el OID deseado.

También tiene la capacidad de monitorizar diferentes tipos de datos al mismo tiempo en una sola gráfica. Un ejemplo sería el número de conexiones TCP en un servidor contra el tráfico total descargado. También tiene la capacidad de publicar los resultados a una página Web en tiempo real.



2. HERRRAMIENTAS SOFTWARE PARA LA GENERACIÓN DE TRÁFICO.

2.1 TFGEN

Este software permite generar tráfico en la red de área local (LAN).

1 Es una aplicación Windows32 GUI válida.

2 Utiliza las librerías de Sockets de Windows.

3 Está diseñado para trabajar bajo redes TCP/IP únicamente. (Utiliza UDP. Debido a esto, se requiere al menos un nodo IP de destino).

4 Tiene la capacidad de generar tráfico multicast. (A partir de la versión 0.4).

5 Puede generar tráfico con patrones específicos. (A partir de la versión 0.6).

6 TFGEN genera tráfico hacia el destino a una tasa de un octeto cada 10 ms. (en el modo "continuo" y "constante")

7 Su uso en plataformas Windows, es recomendado para Windows NT/2000; puede ser utilizado bajo Plataformas Windows 9x, pero la configuración de utilización especificada en el programa no se ajustará a la utilización real de manera exacta; por eso se recomienda su uso estrictamente para laboratorio.

2.1.1 Guía de usuario.

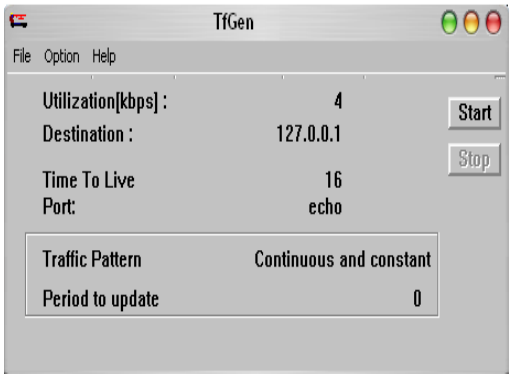
- Primero se debe seleccionar el nodo IP de destino, teniendo en cuenta la capacidad del nodo.
- El siguiente paso consiste en la elección de la utilización en Kbps.

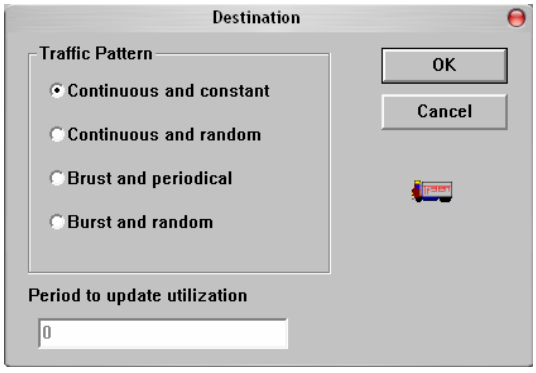
- Luego, se selecciona en el menú File (archivo) la opción Start, y así la herramienta comienza a generar tráfico.

- Como recomendación al usuario de plataformas Windows 9x, se recomienda que seleccione el menú option (opción)/Traffic Pattern (patrón de tráfico) en la ventana de dialogo que aparece se debe seleccionar "Burst and Periodical" (Ráfaga de impulsos) y fijar en 10 la opción "Period to update utilization".

Este tipo de patrón de tráfico, permite generar tráfico de ancho de banda lleno cada "Period to update utilization" (período de actualización de utilización), p Ej. Si este periodo es 10, la herramienta genera tráfico de ancho de banda lleno cada 100ms (1=10ms).

- El patrón de tráfico por defecto "Continuous and Constant" es equivalente al "Burst and Periodical" mientras el periodo de actualización sea 1. Según las pruebas realizadas por el diseñador del software las plataformas Windows 9x no pueden actualizar a una tasa tan alta. Esto indica que el desempeño del manejo de temporización de las plataformas Windows 9x no es adecuado.





2.2 LAN TRAFFIC V2

LanTraffic V2 es un software de herramientas de prueba, que permite generación de tráfico

UDP y TCP en una red IP.

- Este generador de tráfico puede ser configurado con un gran número de parámetros diferentes y soporta hasta 32 conexiones IP simultáneas (TCP o UDP). Este software permite realizar evaluación unitaria y con carga, usando diversas formas de tráfico.

- Esta herramienta esta compuesta de dos partes: el emisor y el receptor.

- La etapa del emisor genera más de 16 conexiones IP simultáneas, mediante dos modos de operación diferentes: unitario y automático.

- En el modo de prueba unitario, el usuario puede seleccionar la fuente del generador de tráfico, y configurar variables como el tamaño de paquete, y el retardo Inter-paquetes para cada conexión.

- En el modo de prueba automático, el usuario selecciona una ley matemática para la conexión, el tiempo de arranque del generador y otra ley matemática para el volumen de datos que será enviado.

- La etapa del receptor recibe tráfico IP (más de 16 conexiones simultáneas). Cada conexión puede ser configurada en un modo de operación: echoer, absorber y file absorber (en este modo los datos recibidos son grabados en un archivo definido por el usuario.

- Es posible generar tráfico TCP y UDP con muchos patrones de carga diferentes y medir el RTT (Round trip time) en cada conexión, esto permite a LanTraffic ser usado en mediciones de desempeño de redes IP.

2.2.1 Funciones principales. Las principales funciones ofrecidas por los menús de este software son:

- Permite la configuración de parámetros TCP como el tamaño de buffer y el tamaño de ventana para la pila de protocolos TCP/IP de Microsoft.

- Las estadísticas de tráfico IP globales como el número de conexiones activas, el número de conexiones fallidas, el throughput de emisión y de recepción total, etc. son mostradas continuamente.

- Los parámetros generales de conexión IP para el emisor son:

- Dirección IP
- Número de puerto
- Protocolo (TCP o UDP)

- Los parámetros para el modo de prueba unitario son:
- Generación de datos de tráfico (hay tres opciones):

- Ley matemática: uniforme, exponencial o pareto.
- Contenido del generador de paquetes: fijo, aleatorio, alternado, o incremento / decremento.
- Nombre del archivo a ser enviado.

- Tamaño de paquete: fijo, aleatorio, alternado, o incremento / decremento.

- Retardo Inter-paquete: fijo, aleatorio, alternado, o incremento / decremento.

- Opción RTT

- Parámetros para el modo de pruebas automático: este modo de trabajo permite que todas las conexiones habilitadas sean generadas simultáneamente siguiendo una ley de

“tiempo de inicio de generación de conexiones” y una ley de “volumen de datos a generar”.

- Selección de las conexiones IP (de 1 a 16).
- Ley de tiempo de inicio de generación de conexiones: uniforme o exponencial.
- Ley de volumen de datos a generar: uniforme, exponencial o pareto.

- Estadísticas de tráfico para el emisor

- Throughput instantáneo: el Throughput desplegado es un Throughput de **“aplicación”.** En ciertos instantes puede ser diferente al Throughput de red física ya que los datos pueden fragmentarse y ser simplificados en varios niveles de sistema.

- Datos enviados (paquetes UDP o bytes TCP).

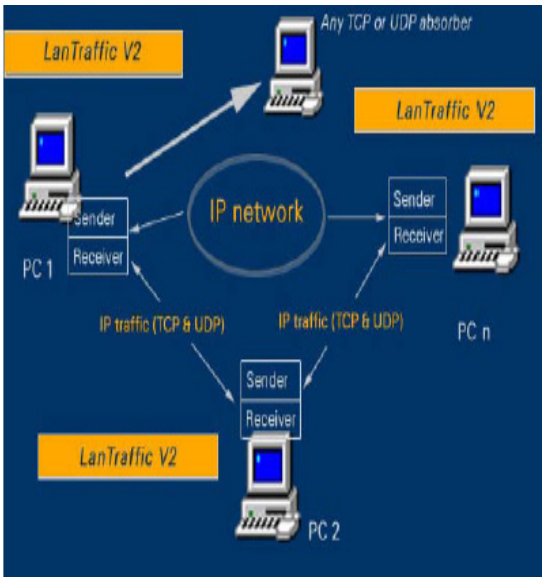
- Los parámetros generales de conexión IP para el receptor son:

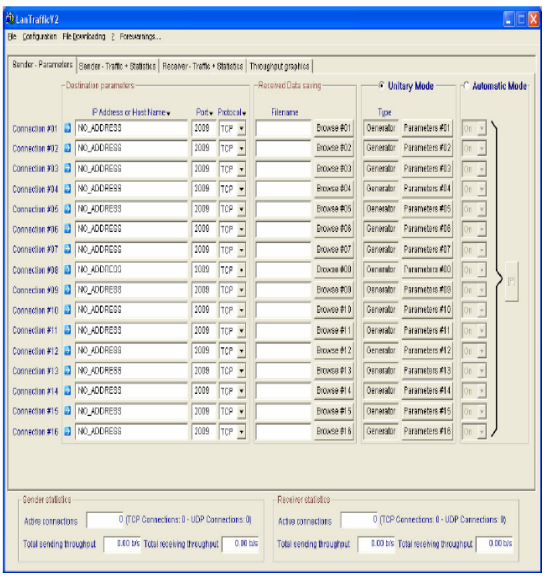
- Dirección IP
- Numero de puerto
- Protocolo (TCP o UDP)
- Modo de recepción seleccionado: echoer, absorber, file absorber o deshabilitado.

- Estadísticas de tráfico para el Receptor

- Throughput instantáneo

- Datos recibidos (paquetes UDP o Bytes TCP)
- Datos repetidos (echoed) (paquetes UDP o Bytes TCP), si el modo echoer esta activo.
- Errores de secuencia de numeración.





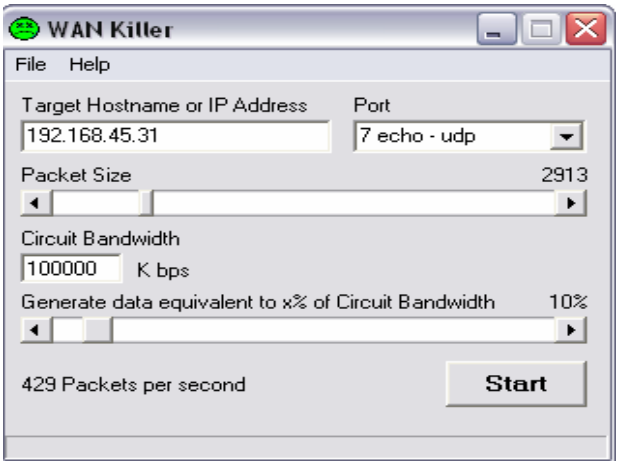
2.3 WAN KILLER

- Genera tráfico aleatorio y permite fijar el ancho de banda del segmento y el porcentaje de carga necesario; también permite ajustar el tamaño del paquete.

- Se debe seleccionar el protocolo de transporte (TCP o UDP) y el número del puerto; a partir de una lista o introducirlo manualmente. También se puede agregar un número de puerto y protocolo.

- Al usar el Puerto 7 (Echo) se genera tráfico en ambas direcciones, todo el trafico que es recibido por la tarjeta del dispositivo, será enviado de regreso a WAN Killer.

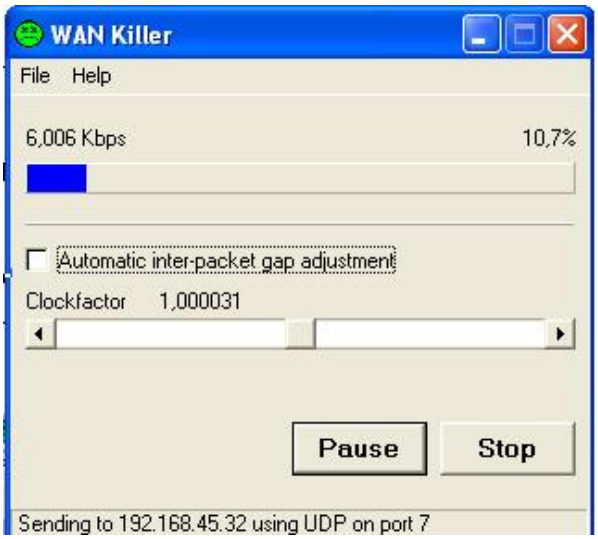
- Por el contrario al usar el puerto 9 (Discard) todo el tráfico que es recibido, es descartado y por lo tanto la carga de tráfico es en una sola dirección.



2.3.1 Número de paquetes. El programa siempre intentara generar el porcentaje de ancho de banda fijado al tamaño establecido; pero si el tamaño del paquete es grande, entonces se enviaran menos paquetes; un tamaño de paquete pequeño, permite generar más paquetes.

El porcentaje de ancho de banda del circuito se puede variar desde 0 hasta 150%.

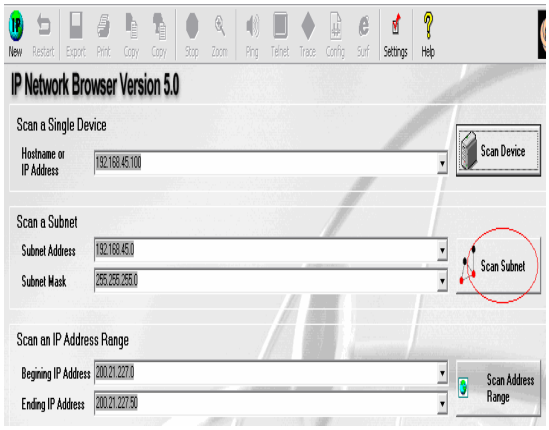
También se puede ajustar el espacio Inter-paquete por medio de un factor de reloj; al variar este factor también se afectará la cantidad de tráfico generado.

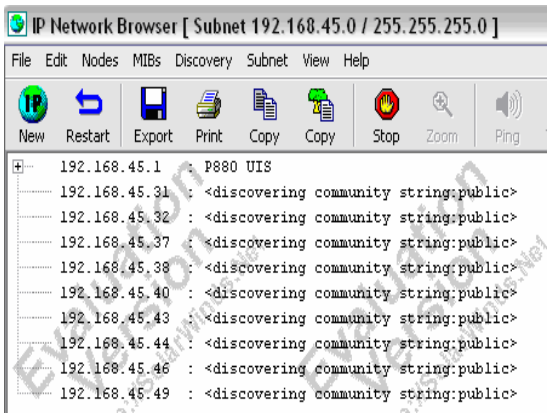


2.4 **IP NETWORK BROWSER** (como generador Broadcast)

Esta herramienta se encuentra dentro del paquete de Network Discovery de Solarwinds y la descripción que se presenta es representativa de los demás programas utilizados para la generación de broadcast ARP que ya han sido mencionados.

Estas herramientas necesitan que el administrador indique el rango de direcciones IP a ser consultado, esta característica y el comportamiento del barrido realizado son mostrados en las siguientes gráficas.





Se puede generar una gran cantidad de paquetes broadcast ejecutando el programa IP Network Browser de Solarwinds, realizando un sondeo a la dirección de la subred donde está el Switch.

El software consulta todas las direcciones IP de los Host según la máscara de subred prefijada, para realizar el mapeo de la subred; para realizar esta tarea la aplicación produce una gran cantidad de tráfico broadcast que podemos aprovechar en nuestras pruebas de laboratorio.

ANEXO E. Especificaciones de los equipos utilizados.

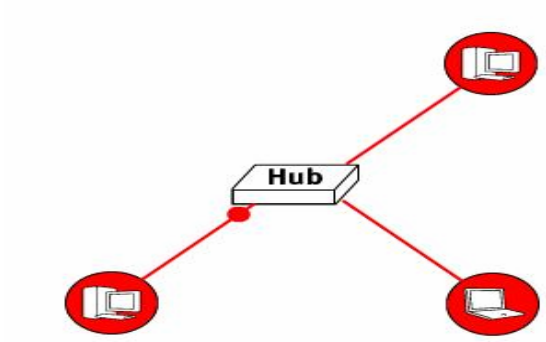
A continuación se listan las especificaciones y referencias de los equipos utilizados en la sala de redes de la escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones de la Universidad Industrial de Santander para la puesta en marcha de las pruebas del proyecto.

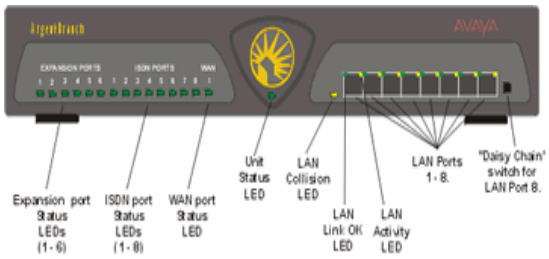
1. HUB AVAYA ARGENT BRANCH

Un hub es simplemente un aparato que repite las señales recibidas. Este no sabe que computadores están conectados a él, y tampoco hace ninguna clase de procesamiento de red basado en el computador origen o destino. Los Hubs son primordialmente usados como aparatos de bajo precio que permiten adherir más computadores a una red.

Pero, a medida que se incrementa el número de computadores, también se incrementa el tráfico innecesario en la red. Utilizar un hub para conectar una red no es muy seguro, ya que cualquier computador puede ser configurado para "escuchar" los mensajes que son transmitidos. Un hub no permite compartir automáticamente una conexión de Internet, aunque puede ser posible si se corre el software de Compartimiento de Conexión de Internet en un computador con conexión de alta velocidad.

Es preferible conectar la red con un switch en vez de un Hub, especialmente desde que los Switches se han vuelto más baratos. Dentro de los equipos de red presentes en el laboratorio de la escuela se encuentra el HUB AVAYA ARGENT BRANCH cuyas características son:





- Posee 8 puertos LAN.
- Posee dos slots para módulos ISDN.
- Puede conectar 2 LANS - 10 MHz y otra de 100 MHz, estas LANS son independientes. El dispositivo interconecta las dos redes. Las dos LANS son 10MHz 192.168.42.1 y 100MHz 192.168.43.1 por defecto. La LAN conectada es determinada por la velocidad de la tarjeta de red del PC, no por el puerto físico del HUB al cual se conecte.
- Ofrece llamada en conferencia –63 usuarios en una conferencia simple, o 21 conferencias en grupos de tres personas.
- Soporta hasta 30 llamadas de datos simultáneas.
- Soporta el modulo de compresión de voz (VCM5/10/20.
- Soporta el modulo Modem2.
- Soporta VoIP.
- Permite realizar sondeos SNMP, los cuales son dirigidos a la dirección IP de 10 MHz.

Este es un dispositivo plug and play por lo que no necesita ninguna configuración.

2. SWITCH AVAYA CAJUN P333R

Cajun P333R es una solución bastante sólida y flexible. Puede alcanzar hasta 240 puertos con su almacenamiento máximo, que es de 10 switches. Avaya ofrece en caso de necesitarse un número mayor de puertos, como recurso opcional, un módulo adicional de 16 puertos y, de esta manera, posibilita que el número máximo pueda llegar a 400. Otra característica bastante interesante, y que ciertamente se debe resaltar, es que Cajun P333R permite un gran número de VLANs (llega a soportar hasta 3.071).

Para la administración y el monitoreo de la red, el switch de Avaya tiene diversos mecanismos como Cajun View, Cajun P330 Manager y Command Line Interface (CLL), entre otros. La administración vía Web, es muy simple y útil, ya que incluye valiosa información para los administradores de red, como multicast, estadísticas de pérdida de paquetes y visión general del almacenamiento, entre otros.

El dispositivo central del esquema de prueba utilizado fue este switch cuyas características más importantes son:

- 24 10/100 Mbps ports + 1 Expansion Slot + 1 Stacking Slot
- Quality of Service (QoS)
- 802.1p Priority support per port.
- 802.1Q VLAN and Port based VLANs
- Congestion Control

- LAG Redundancy

System Overview

The Cajun P333 Stackable Switching System provides modular functionality and port density, carrier-level reliability and building-block simplicity at a stackable switch price. The modularity and power give you the basis for building a network from the closet to the backbone.

Cajun P333 was designed for convergence — the modular features and ease of expansion, in combination with CajunRules policy management and full standards compliance, provide the infrastructure for converged networks.

Key Benefits

• Modular functionality at a stackable price
• Stack managed as a single entity (up to 10 units in a stack)
• Scaleable architecture - both ports and performance added when required
• Multi-technology Solution in the same stack
• Multilayer switching (layers 2, 3, 4) for policy based voice and video networks of tomorrow
• No Single Point of Failure - redundancy at stack, LAG and port levels
• Standards Compliance - interoperability with existing network environments
• Total Cost of Ownership reduced

Standards Supported (on Layer 1 and 2)

• IEEE 802.3x Flow Control on all ports
• IEEE 802.1Q/p VLAN and Priority Tagging on all ports
• IEEE 802.1D Spanning Tree protocol
• IEEE 802.3z Gigabit Ethernet ports
• IEEE 802.3ad LAG protocol
• IETF MIB-II, Bridge MIB
• IETF RMON and SMON

3. EQUIPOS DE CÓMPUTO

Se utilizaron 9 equipos de cómputo para las pruebas, todos con las mismas características:

- Fabricante: Dell Computer corporation
- Modelo: Dell Optiplex Gx 260
- Procesador: Pentium 4 de 2.4 Ghz
- Motherboard: Intel D845GBV
- Memoria RAM: 512 MB DDR SDRAM
- Disco duro: 40GB ATA/100
- Puertos: 6 puertos USB 2.0, 1 puerto serial, 1 puertos paralelo, 2 puertos PS/2, and 1 salida de monitor análoga.
- Tarjeta de red: Integrated gigabit network card
- Sistema operativo: Windows XP Professional edition.