Sistema multi-agente para el monitoreo de tráfico LAN y recursos usados por los equipos*[1]

[Multi-agent system for monitoring LAN traffic and resources used by equipment]

Óscar Eduardo MARÍN OSSA[2], *Julio César GÓMEZ CASTAÑO*[3]

Recibo: 08.05.2010 - Ajuste: 11.10.2011 - Ajuste: 10.02.2011 - Aprobación: 10.06.2011

Resumen

Hay empresas que hacen poca supervisión sobre el ancho de banda que consumen por cada servicio y los recursos de hardware que utilizan, propiciando ineficiencia y dificultad en la detección de fallas. Por ello, se inició el desarrollo de un sistema multi-agente que permitiera la obtención local de los datos del tráfico LAN por servicios y recursos utilizados en cada uno de los equipos de red, cuyos datos se enviarán a la aplicación principal, que permitiera su consulta y gestión. La metodología UPSAM para desarrollo de aplicaciones con agentes permitió la integración del proceso y los modelos específicos para el análisis y diseño de sistemas de agentes de manera iterativa e incremental. Una vez recorrido todo el camino, se concluye que se puede desarrollar aplicaciones para la gestión de redes utilizando sistemas multi-agente por medio de la plataforma Jade. Con el sistema multi-agente distribuido se logra disminuir la carga del servidor para el monitoreo de equipos y con la base de datos central se pueden obtener estadísticas de los equipos monitoreados. Las alarmas del sistema multi-agente

* Modelo para citación de este artículo científico
MARÍN OSSA, Óscar Eduardo y GÓMEZ CASTAÑO, Julio César (2011) Sistema multi-agente para el monitoreo de tráfico LAN y recursos usados por los equipos. En: Ventana Informática. No. 24 (ene-jun., 2011). Manizales (Colombia): Universidad de Manizales. p. 57-76. ISSN: 0123-9678

1 Artículo proveniente del trabajo de grado, realizado por el primer autor para optar por su título de Ingeniero de Sistemas y Telecomunicaciones, en la Universidad de Manizales, bajo la dirección del Ing. Gómez.

2 Ingeniero de Sistemas y Telecomunicaciones. Correo electrónico: oscared29@hotmail.com

3 PhD (c) en Ingeniería Informática, Especialista en Telecomunicaciones e Ingeniero de Sistemas. Docente, Facultad de Ciencias e Ingeniería, Universidad de Manizales. Correo electrónico: jgomez@umanizales.edu.co

ayudan a la administración de los recursos de los equipos y de la red.

Palabras clave*: Gestión, Monitoreo, Tráfico, SNMP, Hardware, Sistema multi-agente.*

Abstract

Some companies make little supervision on the bandwidth consumed by each service and hardware resources used, leading to inefficiency and difficulty in troubleshooting. Thus, began the development of a multi-agent system that would allow local procurement of LAN data traffic for services and resources used in each of the network equipment and sent to the host that would allow for consultation and management. The UPSAM methodology for application development agents allowed the integration of the process and specific models for the analysis and design of agent systems in an iterative and incremental. Once it all the way, we end that you can develop applications for managing networks using multi-agent systems using the Jade platform. With the distributed multi-agent system is able to reduce server load for monitoring equipment and the central data base can provides statistics of monitored equipment. Multi-agent system alerts help management of computer resources and network.

Keywords*: Management, Monitoring, Traffic, SNMP, Hardware, Multi-agent system.*

Introducción

La gestión y monitoreo de tráfico LAN permite conocer el comportamiento real de los valores de la red que cada vez se vuelven más complejos y exigentes soportando más aplicaciones y servicios, también mejora la capacidad de planeación y dimensionamiento ya sea grande o pequeña, así como la medición de su uso y la detección de fallas.

Es importante el uso de un sistema automático de gestión y monitoreo de red que permita a las empresas mantenerse al tanto de lo que ocurre en tiempo real, ayudando así a la administración de la misma y a la toma de decisiones.

Los agentes son entidades inteligentes que pueden ser clasificados por su papel o por sus objetivos en un sistema multi-agente (SMA), siendo este un grupo de agentes que se ejecutan en un entorno para la

solución de uno o varios problemas. Las principales características de los agentes son la movilidad, que les permite transportarse al entorno en el que van a actuar, autonomía que les permite ejecutar acciones de acuerdo a cada servicio lo que perciben de su entorno, distribución que les permite dividir la carga administrativa en un sistema y la comunicación que les permite entrar en contacto unos con otros.

El proyecto propuesto en este documento pretende utilizar los agentes y el SMA para monitorizar el tráfico LAN y los recursos utilizados por los equipos de una red haciendo uso de sus principales características para lograr un sistema distribuido, autónomo, estable y eficaz. También se pretende utilizar el Protocolo Simple de Administración de Red (SNMP) para las comunicaciones, que es ideal en este tipo de sistemas ya que entre sus componentes claves están los agentes, haciendo de su uso algo simple, transparente y sencillo.

Tanto el uso de agentes como el uso del protocolo SNMP en conjunto con sus indiscutibles ventajas contribuirán en el continuo desarrollo de este tipo de sistemas, así como el desarrollo de sistemas para nuevos usos en diferentes ambientes.

La metodología UPSAM hará posible el proceso de ingeniería de software, describiendo el análisis basado en un paradigma orientado a agentes y la posibilidad de representación para llegar a fases de implementación en plataformas Jade (Java Agent Development Framework).

1. Marco teórico

1.1 Simple Network Management Protocol, SNMP

SNMP en sus distintas versiones, de acuerdo con Huidobro (s.f.), es un conjunto de aplicaciones de gestión de red que emplea los servicios ofrecidos por TCP/IP, protocolo del mundo UNIX, y que ha llegado a convertirse en un estándar. Surge a raíz del interés mostrado por la IAB (*Internet Activities Board*) en encontrar un protocolo de gestión que fuese válido para la red Internet, dada la necesidad del mismo debido a las grandes dimensiones que estaba tomando. Los tres grupos de trabajo que inicialmente se formaron llegaron a conclusiones distintas, siendo finalmente el SNMP (RFC 1098) el adoptado, incluyendo éste algunos de los aspectos más relevantes presentados por los otros dos: HEMS (*High-Level Management System*) y SGMP (*Simple Gateway Monitoring Protocol*).

Para el protocolo SNMP la red constituye un conjunto de elementos básicos -Administradores o *Management Stations* - ubicados en el/los equipo/s de gestión de red y Gestores Network Agentes (elementos pa-

sivos ubicados en los nodos host, routers, módems, multiplexores, etc. a ser gestionados), siendo los segundos los que envían información a los primeros, relativa a los elementos gestionados, por iniciativa propia o al ser interrogados (*polling*) de manera secuencial, apoyándose en los parámetros contenidos en sus MIB (*Management Information Base*). Su principal inconveniente es el exceso de tráfico que se genera, lo que lo puede hacer incompatible para entornos amplios de red; por contra CMIS/CMIP (*Common Management Information Service/Protocol*) de OSI ofrece un mejor rendimiento y seguridad, estando orientado a la administración de sistemas extendidos.

El protocolo de gestión SNMP facilita, de una manera simple y flexible, el intercambio de información en forma estructurada y efectiva, proporcionando significantes beneficios para la gestión de redes multivendedor, aunque necesita de otras aplicaciones en el NMS que complementen sus funciones y que los dispositivos tengan un software Agente funcionando en todo momento y dediquen recursos a su ejecución y recogida de datos.

1.1.1 Amenazas a la seguridad. El protocolo SNMP proporciona mecanismos para el acceso a un almacén de información jerárquica compuesta por un conjunto de variables. Se distinguen dos tipos distintos de acceso a dicha información: un acceso para lectura que permite consultar los valores asociados a cada una de las variables y un acceso para escritura que permite modificar dichos valores.

Los mensajes de la primera versión del protocolo incluyen una cadena de caracteres denominada nombre de comunidad que se utiliza como un sencillo mecanismo de control de acceso a la información. Los agentes que implementan dicha versión del protocolo disponen generalmente de dos comunidades, o conjuntos de variables (no necesariamente disjuntos), identificadas por un nombre de comunidad configurable por el administrador del sistema. Una de dichas comunidades recibe el nombre de comunidad pública, y sus variables pueden ser accedidas sólo para lectura. Por el contrario los valores asociados a las variables que componen la otra comunidad, denominada comunidad privada, pueden ser modificados.

Toda la seguridad proporcionada por el sistema se basa en el hecho de que es necesario conocer el nombre asignado a una comunidad para conseguir el acceso a la información proporcionada por sus variables. El nivel de protección ofrecido por la versión original del protocolo es, por tanto, muy débil. Más aún si se tiene en cuenta que los nombres de comunidad incluidos en los mensajes del protocolo SNMP viajan por la red en texto plano y por consiguiente pueden ser obtenidos como resultado de ataques pasivos (escuchas malintencionadas).

1.2 Sistemas multi-agente

Corresponde a la rama de la Inteligencia Artificial Distribuida, IAD; que estudia el comportamiento de agentes inteligentes que resuelven un problema de manera cooperativa, comentan Quintero, Rueda y Ucrós (s.f.). Un SMA (Sistema Multi-agente) está constituido por un conjunto de entidades inteligentes llamadas agentes que coordinan sus habilidades para la resolución de problemas individuales o globales. Estos sistemas considerados como un todo, exhiben características particulares, que se presentan a continuación:

- Organización Social. Es la manera como el grupo de agentes está constituido en un instante dado. La organización social está relacionada con la estructura de los componentes funcionales del sistema, sus características, sus responsabilidades, sus necesidades y la manera como realizan sus comunicaciones. Esta organización puede ser estática o dinámica, dependiendo de las funciones o tareas de cada agente.
- Cooperación. En un SMA existen dos tipos de tareas que deben ser realizadas: las tareas locales y las tareas globales. Las tareas locales son las tareas relacionadas con los intereses individuales de cada agente y las tareas globales son las tareas relacionadas con los intereses globales del sistema.
- Coordinación. La coordinación entre un grupo de agentes les permite considerar todas las tareas a realizar y coordinarlas para no ejecutar acciones no deseables.
- Negociación. Para que los mecanismos de cooperación y coordinación sean exitosos en un sistema de agentes que actúan interdependientemente, debe existir un mecanismo adicional, por medio del cual, los integrantes de un sistema se puedan poner de acuerdo cuando cada agente defiende sus propios intereses, llevándolos a una situación que los beneficie a todos teniendo en cuenta el punto de vista de cada uno.
- Control. El control es el mecanismo básico que provee apoyo para la implementación de mecanismos de coordinación en un SMA.

1.2.1 Propiedades. Las siguientes son propiedades de los sistemas multi-agente:

- Autonomía: capacidad de actuar sin intervención humana directa o de otros agentes.
- Sociabilidad: capacidad de interaccionar con otros agentes, utilizando como medio algún lenguaje de comunicación entre agentes.

- Reactividad: un agente está inmerso en un determinado entorno (hábitat), del que percibe estímulos y ante los que debe reaccionar en un tiempo preestablecido.
- Iniciativa: un agente no sólo debe reaccionar a los cambios que se produzcan en su entorno, sino que tiene que tener un carácter emprendedor y tomar la iniciativa para actuar guiado por los objetivos que debe de satisfacer.

1.2.2 Comunicación entre agentes. En los sistemas multi-agente la comunicación entre agentes es muy importante para llevar a cabo variados objetivos. A continuación se muestran las diferentes formas de comunicación:

- No hay comunicación. Los agentes pueden interactuar sin comunicarse, infiriendo las intenciones de otros agentes. Este modelo de comunicación es el que consigue mejores resultados cuando los objetivos de los agentes no están interrelacionados y por lo tanto no pueden entrar en conflicto.
- Comunicación primitiva. En este caso la comunicación está restringida a un número de señales fijas con una interpretación establecida de antemano.
- Arquitectura de pizarra. Se usa en la inteligencia artificial como una manera de compartir memoria y conocimiento. Los agentes pueden escribir mensajes, dejar resultados parciales o encontrar información en una pizarra que todos saben dónde está. Se usa esta arquitectura para intercambiar información entre subsistemas de agentes o para modelar el intercambio entre los distintos módulos que componen la estructura de un agente.
- Intercambio de planes e información. Dos agentes pueden intercambiarse sus respectivos planes, de esta manera cada uno puede adaptar sus estrategias. Esto presenta algunos inconvenientes como el alto coste computacional del intercambio.
- Intercambio de mensajes. Consiste en agentes que actúan en respuesta al procesamiento de una comunicación. Las acciones que pueden ejecutar estos agentes son: enviarse mensajes entre ellos mismos, determinar un cambio en el comportamiento como consecuencia de haber alcanzado un nuevo estado después de la comunicación, etc.
- Comunicación de alto nivel. El diálogo entre agentes permite la generación e interpretación de palabras o declaraciones, con el objetivo de comunicar la información que el emisor conoce formada por creencias, compromisos e intenciones, para que así el receptor alcance el mismo estado mental que tiene el emisor.

1.3 Antecedentes

1.3.1 Smoothwall Express. Según Smoothwall (2009), es un sistema cortafuego modular, diseñado para satisfacer las necesidades de seguridad en la red empresarial y de las sucursales. Un diseño modular permite que el sistema de seguridad progrese con las necesidades del negocio, proporcionando la capacidad del ingreso remoto con un alto nivel de seguridad para el personal de campo y proporciona servicios de *firewall*, enrutamiento, NAT, *Logging server*, DNS *Proxy server*, SSH, IDS, gráficos de seguimiento de tráfico de red y VPN basada en IPSEC.

1.3.2 Ciscoworks. El diccionario de Babylon (2009) define CiscoWorks como: «*Serie de aplicaciones de software de administración de internetwork basadas en SNMP. CiscoWorks incluye aplicaciones para controlar el estado del servidor de acceso y del router, administrar los archivos de configuración, y diagnosticar las fallas de la red. Las aplicaciones de CiscoWorks se encuentran integradas con varias plataformas de administración de red basadas en SNMP, incluyendo SunNet Manager, HP OpenView e IBM NetView*».

CiscoWorks (Cisco.com, 2009), solución para la administración de LAN es un conjunto de ponderosas herramientas de administración que simplifican la configuración, administración, monitorización y solución de problemas de redes Cisco. Integra las siguientes capacidades en la mejor solución de su clase para:

- Mejorar la precisión y eficiencia de la operaciones de la red
- Incrementado la disponibilidad general de la red al simplificar la configuración e identificando y solucionando los problemas de red
- Maximizando la seguridad de la red a través de la integración de servicios de control de acceso y la auditoria de cambios a nivel de red.

1.3.3 Aranda IT Asset Management. Consiste, de acuerdo con Aranda (2009), en un portafolio de soluciones que contienen las mejores prácticas para administrar y gestionar la infraestructura IT de una compañía. Estas soluciones permiten tener un inventario completo de los activos, realizar un seguimiento al ciclo de vida de los mismos y auditar y monitorear el uso de software.

1.3.4 3com Network Supervisor. 3Com® Network Supervisor es una aplicación comercial de administración fácil de usar que descubre, asigna y muestra gráficamente enlaces de redes y dispositivos IP, incluyendo teléfonos IP 3Com® y algunos populares productos de otros fabricantes (3Com, 2009).

1.3.5 PRTG Network Monitor. De acuerdo con Paessler (2009), es una potente herramienta comercial y libre con 10 sensores, de monitorización

de la Paessler AG, que asegura la disponibilidad de componentes de red y mide el tráfico y el uso de la red y ahorra costos ayudando a evitar fallos, optimizar conexiones, economizando tiempo de implementación y controlando acuerdos de nivel de servicio (SLA).

Cubre todos los aspectos de la monitorización de redes: monitorización de disponibilidad, monitorización de tráfico y de uso, SNMP, NetFlow, *sniffer* de paquetes y muchos más, al igual que funciones de reporte y de análisis.

1.3.6 Sistema Multi-agente para la Recuperación y Monitoreo de Actividades de Usuarios y Software en los Equipos de una Red. Proviene de una tesis realizada en la Universidad Autónoma de Manizales, que presenta un sistema multi-agente para recuperar y gestionar datos sobre el software instalado y sobre las actividades realizadas en los equipos de una red (López, 2006).

1.3.7 Sistema SCADA para la Administración y Monitoreo de Equipos en una Red de Datos. Corresponde a una tesis realizada en la Universidad de Manizales que presenta un sistema SCADA, utilizando herramientas de Microsoft .NET, para la administración y monitoreo del hardware y el software de los equipos en una red de datos de los cuales sólo hace inventario, usando un modelo cliente/servidor (Montes, 2009).

2. Metodología

El desarrollo del proyecto implicó las fases de Inicio, análisis y diseño, e implementación, en la Universidad de Manizales.

2.1 Inicio

Se profundizó en el Protocolo Simple de Administración de Red (SNMP) y multi-agentes; y se instalaron y configuraron las herramientas necesarias en el PC para llevar a cabo la implementación. Los sistemas operativos elegidos fueron *Windows XP SP3* (cliente) y dos *Linux Ubuntu* 9.10 (uno cliente y el otro servidor). Para el desarrollo de los agentes se instalaron *Java Runtime Environment* (RE) 6 *update* 17, *Java Development Kit* (SDK) 6 *update* 17 y la plataforma para el desarrollo de agentes JADE 3.7. La instalación de la base de datos MySQL era necesaria para la permanencia de los datos generados por los agentes, además de una interfaz de usuario que facilitara la creación, configuración y pruebas de la misma. Se configuró el servicio SNMP en los sistemas operativos para la obtención de los datos de los equipos. Finalmente se adecuaron las librerías *Jpcap* y *WinPcap* 4.1.1 (*Windows XP*) para obtener los datos del tráfico de red LAN.

2.2 Análisis y diseño

De acuerdo con la metodología de la UPSAM, se llevaron a cabo las siguientes actividades:

- Describir el sistema, en este caso, del sistema multi-agente para el monitoreo de tráfico LAN y los recursos utilizados por los equipos de la red.
- Construir los Modelos de Casos de Uso Orientado a Agentes y se identificaron los actores del sistema.
- Definir los Roles y Agentes. Se definieron las principales tareas de cada agente que pueden evidenciarse en el sistema para el monitoreo de tráfico LAN y los recursos utilizados por los equipos de la red.
- Describir el Modelo de Arquitectura y Organización de Agentes
- Modelar los Recursos. Se describieron los objetos y entidades que pertenecen al sistema, incluyendo los agentes. También el entorno exterior y como es percibido por los agentes.
- Describir el Modelo de Comunicación de Agentes. Se describieron por medio de diagramas de secuencia la interacción entre los distintos tipos o roles de agentes.
- Definir el Modelo Interno de Agentes. Se describieron los agentes que se van a instanciar en el sistema y los comportamientos para su funcionamiento.

2.3 Implementación. La implementación del Prototipo con JADE se realizó mediante la utilización de las herramientas mencionadas en la primera fase. La correcta instalación y configuración de dichas herramientas permitió llevar a cabo el proyecto. Se ejecutó el prototipo implementado en la red y en varios equipos de la Universidad de Manizales de manera exitosa.

3. Resultados y discusión

Se logró crear el Sistema Multi-agente para la obtención de datos de los equipos monitoreados por medio del Agente Servicio. En la figura 1 se muestran los datos de hardware recopilados en un equipo de prueba. Los datos obtenidos son:

- El porcentaje de uso de cada núcleo que el procesador tenga.
- La cantidad de memoria del sistema total, usada y libre dada en MB.
- La cantidad de discos duros del equipo así como las capacidades totales, cantidades usadas y libres dadas en GB.

- Datos relevantes de las tarjetas de red como el estado, errores de operación, paquetes de entrada y salida descartados y datagramas descartados.

Figura 1. Datos de los recursos

También se logró el monitoreo del tráfico por servicios, por medio del Agente Sniffer:

- FTP (*File Transfer Protocol*)
- SSH (*Secure Shell*)
- SMTP (*Simple Mail Transfer Protocol*)
- DNS (*Domain Name System*)
- HTTP (*HyperText Transfer Protocol*)
- POP3 (*Post Office Protocol 3*)
- IMAP4 (*Internet Message Access Protocol 4*)
- HTTPS (*Hypertext Transfer Protocol Secure*)

El Agente Sniffer captura todos los paquetes generados por el tráfico de red, de entrada como de salida, seguidamente filtra y separa los mismos por servicio. Finalmente envía los datos recogidos a la base de datos que se encuentra en el servidor. Allí el Agente Gestión se encarga de generar y mostrar las estadísticas. En la figura 2 se muestran los datos del tráfico de red recopilados en un equipo de prueba.

Figura 2. Estadísticas de tráfico por el sniffer de un equipo

Todos los datos obtenidos son integrados en la base de datos MySQL que se encuentra ubicada en el servidor que permite llevar registros continuos e incrementales útiles para la gestión integral de todos los equipos monitoreados. La base de datos contiene toda la información enviada por los agentes en los equipos Windows y Linux:

- Datos de los procesadores
- Datos de las memorias
- Datos de los discos duros
- Datos de las tarjetas de red
- Datos de alertas
- Datos de errores
- Datos del sniffer
- Datos de parámetros

Para la gestión se crea el Agente Gestión, se encarga como el nombre lo sugiere, de la gestión de todos los equipos monitoreados por el Sistema Multi-agente, permite:

- Ver los equipos monitoreados.
- Ver las estadísticas de los procesadores, memorias, discos duros y tarjetas de red.
- Ver las estadísticas del tráfico de red por servicios.
- Ver las alertas generadas por los agentes.
- Ver los errores producidos.

- Administrar usuarios.
- Configuración del Agente Servicio y Agente Sniffer.

En la figura 3 se muestra el menú principal, el cual se accede al ejecutar el Agente Gestión y pasar por el proceso de autenticación.

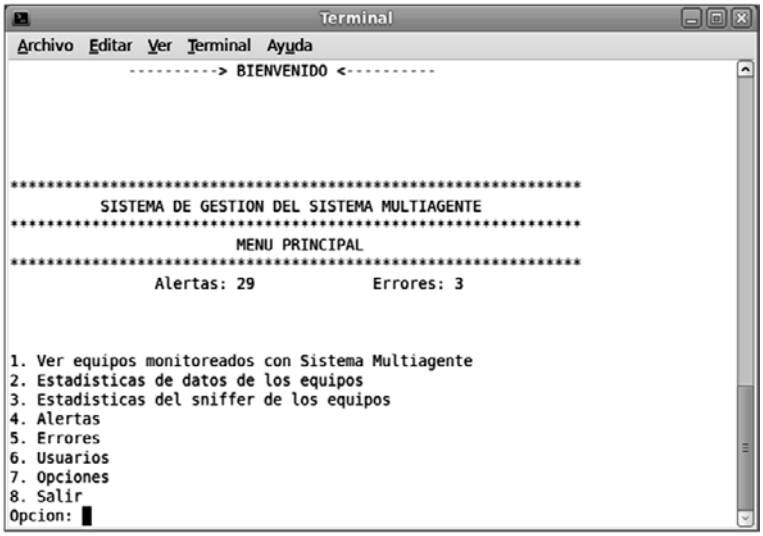

Figura 3. Menú principal del Agente Gestión

Se creó el Sistema Multi-agente para que funcionara de manera distribuida, lo que le da independencia de operación, mayor protección contra errores y disminuye la carga sobre el servidor. En la figura 4 se muestra la distribución del Sistema Multi-agente. Se visualiza el servidor que contiene la base de datos y el Agente Gestión; de manera opcional también puede ejecutar los Agentes Servicio y Sniffer. De igual manera, se visualizan los equipos con sistemas operativos Windows y Linux los cuales ejecutan los Agentes Servicio y Sniffer.

Para la adaptación en el entorno, se incluyó capacidades de configuración por medio del Agente Gestión, de esta manera el administrador encargado puede:

- Cambiar los rangos de alertas para las memorias de todos los equipos en el Agente Servicio.
- Cambiar los rangos de alertas para los discos duros de todos los equipos en el Agente Servicio.
- Cambiar la dirección IP por la cual el Agente Sniffer hará el monitoreo de tráfico de red, en caso de que el equipo objetivo tenga dos o más tarjetas de red.

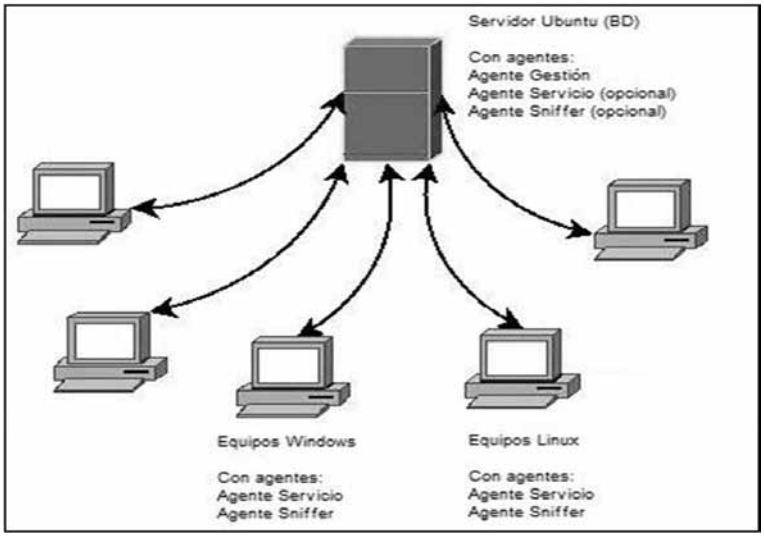

Figura 4. Topología del Sistema Multi-agente

Se logró implementar un sistema para el monitoreo de tráfico y de recursos de los equipos, utilizando un sistema multi-agente, con Java y la plataforma de agentes Jade, a diferencia de varias soluciones que utilizan los lenguajes de programación C, Perl. Las herramientas de desarrollo como Java, Jade, MySQL, librerías Jpcap y el sistema operativo Linux utilizados para la elaboración del proyecto son todos de uso libre excepto el sistema operativo Windows.

Se diseñó el Sistema Multi-agente de manera distribuida, donde cada equipo monitoreado posee sus agentes autónomos. La información generada en cada equipo es enviada y guardada en una base de datos central, ubicada en un servidor para su consulta y gestión. De manera adicional, los agentes también guardan la información de manera local por medio de logs (archivos planos).

Durante todo el proceso de implementación aparecieron diversos problemas, especialmente en la programación con el lenguaje Java y Jade. Sin embargo, ninguno de ellos impidió el cumplimiento de todos los objetivos propuestos. En la plataforma Windows se utilizó la librería SNMP para Java para realizar las consultas requeridas. En la plataforma Linux (Ubuntu 9.10) no funcionó dicha librería, por lo que se recurrió a utilizar comandos SNMP del sistema operativo, llamados desde el agente.

En el siguiente código se muestra la implementación de SNMP en el Agente Servicio para sistemas operativos Windows:

```
InetAddress hostAddress = InetAddress.
getByName(direccionIP);

SNMPv1CommunicationInterface comInterface = new SNM
Pv1CommunicationInterface(versionSNMP, hostAddress,
comunidad);

SNMPVarBindList tableVars = comInterface.
retrieveMIBTable(OIDtoget);
```

Con el comando SNMP: *snmpwalk –c public –v2c localhost 1.3.6.1.2.1.1.25.3.3.1.2* se está solicitando al localhost con la comunidad pública (public) y la versión 2c de SNMP, la consulta del uso del procesador.

En la figura 5 se aprecia el comando SNMP: *snmpwalk –c public –v2c localhost 1.3.6.1.2.1.1.25.2.3.1.2* en el cual se está solicitando al localhost con la comunidad pública y la versión 2c de SNMP, el OID que permite consultar los tipos de almacenamiento que posee un equipo.

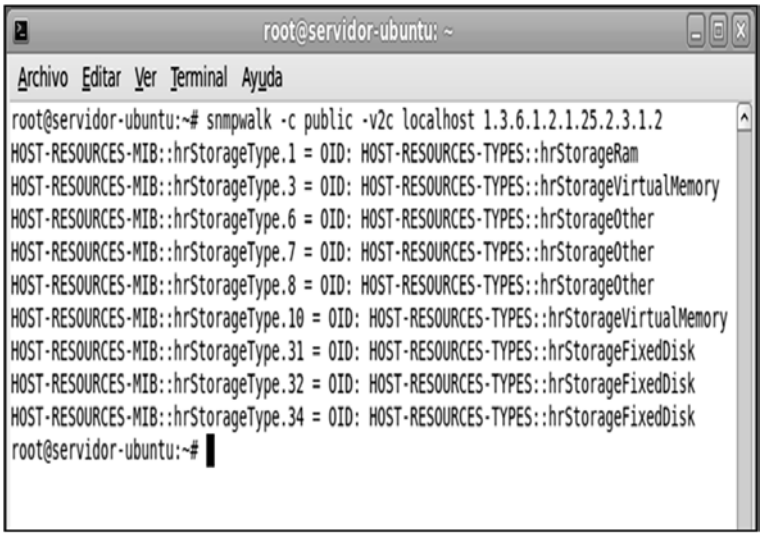

Figura 5. Comando SNMP para obtener los tipos de almacenamiento

Para la autenticación del usuario en el Agente Gestión se hizo uso de las recomendaciones de seguridad del Proyecto OWASP, ya que en el proceso de autenticación, se pueden sufrir ataques como el *SQL Injection*, el cual se probó y permitió el ingreso no autorizado al sistema. Se agregó la recomendación OWASP y el sistema quedó más seguro, prueba de ello fue que el ataque, ya no tuvo éxito. También se utiliza el cifrado de la contraseña de usuario por medio del método MD5. En el

siguiente código se muestra la implementación de la protección contra *SQL Injection*, que hace parte de las recomendaciones de seguridad propuestas por el Proyecto OWASP.

```
String linea = "SELECT * FROM UsuariosGestion WHERE Nom-
bre_Usuario = ? AND Password_Usuario = ?";

PreparedStatement declPreparada = conectar.
prepareStatement(linea);

declPreparada.setString(1, nu);

declPreparada.setString(2, strPW);

ResultSet resultado = declPreparada.executeQuery();
```

En el siguiente código se muestra la implementación del cifrado de la contraseña de usuario, que hace parte de las recomendaciones de seguridad propuestas por el Proyecto OWASP.

```
strPW = encriptacionMD5(strPW);
```

También se realizó uso de seguridad adicional como la entrada segura de la contraseña de usuario, que impide la visualización de la misma mientras se digita; eliminación de la contraseña una vez utilizada, para evitar ataques a la memoria que permitan la obtención de la misma; y el uso del modificador *private* para variables y métodos de todos los agentes, que impide cualquier acceso desde el exterior. En el siguiente código se muestra la implementación de la entrada segura de la contraseña de usuario.

```
System.out.print("Nombre de usuario: ");

String nu = entrada.readLine();

System.out.print("Password: ");

char[] pw = entradaPass.readPassword();
```

En el siguiente código se muestra la implementación del borrado de las contraseñas de usuario, tanto la cifrada como la no cifrada.

```
Arrays.fill(pw, ' ');

strPW = "";
```

En el siguiente código se muestra la implementación del modificador *private*. En este caso para el método *Login*.

```
private boolean Login()

{ … }
```

El Sistema multi-agente genera alarmas de acuerdo a la configuración definida por el administrador, en la que se puede informar del uso medio, alto y crítico de la memoria del sistema (RAM) y los discos duros de cada

equipo monitoreado. En el siguiente código se muestra la implementación de las alarmas. En este caso es para el uso de la memoria RAM.

```
float calcMem = (memUsadaMB * 100)/tamMemMB;
int NivelMedio = 70;
int NivelAlto = 80;
int NivelCritico = 90;

ResultSet resultado = declaracion.executeQuery("SELECT *
FROM ServicioEntrada WHERE Destino_Entrada='Memoria'");


if (resultado.next())
{
NivelMedio = resultado.getInt("Nivel_Medio");
        NivelAlto = resultado.getInt("Nivel_Alto");
        NivelCritico = resultado.getInt("Nivel_Critico");
}
if (calcMem >= NivelMedio && calcMem < NivelAlto)
        reporteAlertas("El uso de la memoria es del "+Ni-
velMedio+"% al
 "+(NivelAlto-1)+"%", "Medio");
else
        if (calcMem >= NivelAlto && calcMem < NivelCritico)
reporteAlertas("El uso de la memoria es del "+NivelAlto+"%
al "+(NivelCritico-1)+"%", "Alto");
        else
                if (calcMem >= NivelCritico)
reporteAlertas("El uso de la memoria es de mas del "+Ni-
velCritico+"%", "Critico");
```

En el Agente Sniffer el monitoreo del tráfico de red se hace de forma cíclica no convencional tal como se muestra a continuación. No se hace uso de ciclos comunes como for, while, do-while.

```
jpcap.loopPacket(1000,new agsniffer());
```

Se pueden visualizar los agentes por medio de la interfaz gráfica de Jade, en la cual se puede observar sus nombres, sus estados y el nombre del equipo en el que se están ejecutando. Esto se logra gracias a la opción *–gui* que se agrega al momento de la ejecución. En la figura 6 se puede apreciar los agentes Gestión, Servicio y Sniffer que son visualizados por medio del entorno gráfico de Jade en un equipo Linux. El

comando utilizado es: *runjade –gui servicio:agservicio sniffer:agsniffer gestión:aggestion*.

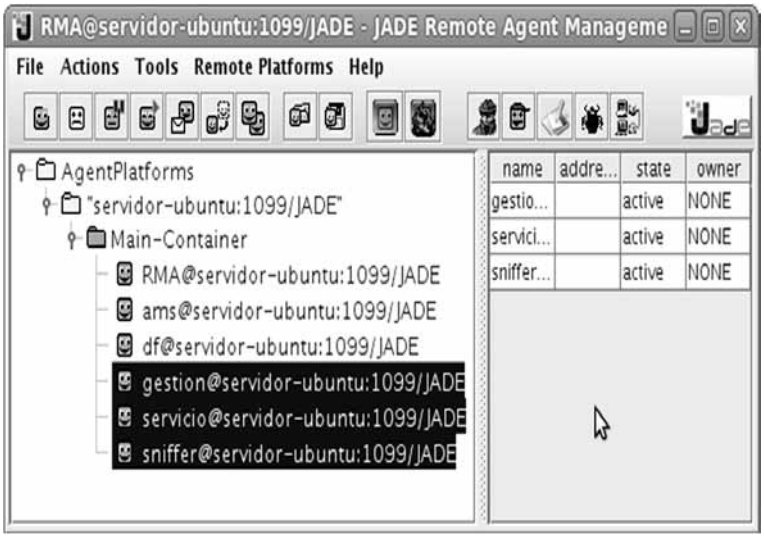

Figura 6. Agentes Gestión, Servicio y Sniffer en entorno Jade

4. Conclusiones

- En equipos con el sistema operativo Windows, la librería Jpcap usa componentes externos para su funcionamiento, para ello se debe instalar también la librería WinPcap para asegurar su correcto funcionamiento así como el del Agente *Sniffer*. Esta situación no ocurre en equipos con el sistema operativo Linux, cuya instalación de Jpcap no requiere de componentes externos.
- Para la obtención de los datos de la tarjeta de red tanto en equipos con el sistema operativo Windows como Linux se utilizó la MIB “HOST-RESOURCES-MIB”, sin embargo también se hizo uso de la MIB “RFC1213-MIB” que proporcionó datos no disponibles en la primera MIB.
- En el Agente *Sniffer* resultó difícil la implementación del código fuente para la selección automática de una tarjeta de red para monitorear debido a que el sistema operativo Windows y especialmente el sistema operativo Linux reportaban otros dispositivos como tarjetas de red. La solución fue filtrar aquellos dispositivos sin una dirección IP, además de filtrar dispositivos con dirección IP v6. De esta manera se logró obtener las tarjetas de red válidas. El paso

final fue la selección del dispositivo en el caso de que un equipo dado tuviera más de una tarjeta de red válida, donde se escogía el primero dado a que en la mayoría de equipos el primer dispositivo es aquel conectado a la red local (LAN), a internet o ambos, y los demás dispositivos son secundarios y/o virtuales creados por programas como *VMware* y *VirtualBox*.

- El uso de un entorno gráfico para la creación de la base de datos MySQL y sus tablas fue de mucha utilidad durante el proceso de implementación, debido a que permitió visualizar con facilidad y rapidez las tablas, sus campos y datos contenidos, lo que a su vez facilitó la detección de errores y cambios necesarios de acuerdo a la evolución del Sistema Multi-agente.
- A pesar de que el lenguaje de programación Java y Jade (basado en Java) son multiplataforma, hay que tener cuidado con las diferencias entre los sistemas operativos, ya que no todo funciona igual entre éstos y no todos los componentes externos (librerías) son multiplataforma también. Como ejemplo, se puede citar la ejecución de agentes en equipos con sistema operativo Windows y Linux. La compilación de código fuente de los agentes genera archivos *.class* que pueden ser ejecutados directamente en Windows, sin embargo, este no es el caso en Linux, donde los archivos compilados deben estar contenidos en un archivo *jar* antes de su ejecución.
- Con el Sistema Multi-agente distribuido se logra disminuir la carga de la red y la del servidor para el monitoreo de equipos y con la base de datos central se pueden obtener estadísticas de los equipos monitoreados.
- Las alarmas del Sistema Multi-agente ayudan al administrador a manejar los recursos de los equipos y de la red, ya que le informan automáticamente el estado de los mismos, manejando y personalizando tres niveles de alerta.

Bibliografía

3COM (2009). 3Com Network Supervisor. [en **línea].** Estados Unidos: 3Com. <http://www.3com.com/prod/es_LA_AMER/detail.jsp?tab=features&sku=3CR15100> [consulta: 20/04/2009]

ÁLVAREZ, C. y SOTO P., A. F. (s.f.).Diseño de un Sistema Multi-agente para un Software de Simulación de Celdas de Producción Robóticas [en línea]. Valdivia (Chile): Rama Estudiantil IEEE de la Universidad Austral de Chile. 6 p. <http://ewh.ieee.org/sb/chile/uach/archivos/postgrado4.pdf> [consulta: 20/11/2010]

ARANDA Software (2009). IT Asset Management. [en **línea]. Estados Unidos: Aranda Software.** <http://www.arandasoft.com/itam.php> [consulta: 20/04/2009]

CASTILLO, A. (2004). Modelos y Plataformas de Agentes Software Móviles e Inteligentes para Gestión del Conocimiento en el Contexto de las Tecnologías de la Información. Tesis doctoral. Madrid: Universidad Pontificia de Salamanca. 515 p.

CISCO Systems Inc. (2009). CiscoWorks LAN Management Solution [online]. Estados Unidos: Cisco Systems Inc. <http://www.cisco.com/en/US/products/sw/cscowork/ps2425/index.html> [consulta: 20/04/2009]

DE LA CRUZ, C. R.; BARRERA CÁMARA, R. A.; PÉREZ CRUZ, J. A. y FLORES HERNÁNDEZ, J. A. (2007). Detección de anormalidades en las redes LAN mediante la implementación de un PCA [en línea]. Villahermosa (Tabasco, México): Universidad Juárez Autónoma de Tabasco (18/06/2007), En: GARCÍA MOLINA, H. (2007). Avances en Informática y Sistemas Computacionales, Tomo II. 1. Redes y Sistemas distribuidos. 9 p. <http://www.conais.ujat.mx/historico/conais2007/descargas/_1.pdf> [consulta: 23/11/2010]

DEPARTMENT OF COMPUTER SCIENCE (s.f.). Java SNMP Package [online]. Philadelphia (PA, USA): Department of Computer Science, Drexel University. <http://gicl.cs.drexel.edu/people/sevy/snmp/docs/index.html> [consulta: 09/07/2009]

FERNANDEZ, J. C.; CORRALES, J. A. y OTERO, A. (2007). La seguridad en la familia de protocolos SNMP. [en línea]. Madrid (España): RedIRIS. <http://www.rediris.es/difusion/publicaciones/boletin/50-51/ponencia16.html> [consulta: 14/01/2010]

GUERRERO CASTELEIRO, A. (2007). Especificación del comportamiento de gestión de red mediante ontologías. Tesis doctoral. Madrid (España): Universidad Politécnica de Madrid. Escuela Técnica Superior de Ingenieros de Telecomunicación. Departamento de Ingeniería de Sistemas Telemáticos. 358 p.

HUIDOBRO, J. M. (s.f.). SNMP. Un protocolo simple de gestión [en línea]. Madrid (España): Colegio Oficial de Ingenieros de Telecomunicación. <http://www.coit.es/publicac/publbit/bit102/quees.htm> [consulta: 19/04/2009]

ISAZA ECHEVERRY, G. A. (2008). Modelo de Detección de Intrusos basado en Sistemas Multi-agente y representaciones ontológicas de firmas y reglas. Proyecto de investigación. Madrid (España): Universidad Pontificia de Salamanca. Facultad de Informática. 190 p.

LLAMAS BELLO, C. (2000). Introducción a los Agentes y Sistemas Multi-agente [en línea]. Valladolid (España): Universidad de Valladolid. 51 p. <http://www.infor.uva.es/~cllamas/MAS/MAS.pdf> consulta: 20/11/2010]

LÓPEZ, C. A. (2006). Sistema multi-agente para la recuperación y monitoreo de actividades de usuarios y software en los equipos de una red. Trabajo de grado (Ingeniero de Computación). Manizales (Colombia): Universidad Autónoma de Manizales. Ingeniería de Computación.

MONTES BOTERO, M. E. (2009). Sistema SCADA para la administración y monitoreo de equipos en una red de datos. Trabajo de grado (Ingeniero de Sistemas y Telecomunicaciones). Manizales (Colombia): Universidad de Manizales. Facultad de Ingeniería.

ORACLE (s.f.). MySQL. [en línea]. Redwood Shores (CA, USA): Sun Microsystems. <http://www.sun.com/software/products/mysql/> [consulta: 09/03/2010]

OWASP (2010). Java Project. Java Security Overview [online]. Columbia (MD, USA): OWASP Foundation. (actualizado: 30/08/2010) <http://www.owasp.org/index.php/Category:OWASP_Java_Project> [consulta: 07/09/2010]

PAESSLER (2009). PRTG Network Monitor V7 [en línea]. Alemania: Paessler. <http://www.es.paessler.com/prtg/> [consulta: 20/04/2009]

QUINTERO, A.; RUEDA RODRIGUEZ, S. y UCRÓS C., M. E. (2009). Agentes y Sistemas Multiagente: Integración de Conceptos Básicos [en línea]. Bogotá (Colombia): Universidad de los Andes. <http://agamenon.uniandes.edu.co/yubarta/agentes/agentes.htm> [consulta: 20/04/2009]

SÁNCHEZ PACHECO, C. A. y FERMÍN, J. R. (2009). Vulnerabilidad del protocolo Mysql en redes LAN bajo plataforma LINUX [en línea]. En: Télématique: revista Electrónica de Estudios Telemáticos, Vol. 8, No. 1. Maracaibo (Zulia, Venezuela): Universidad Rafael Belloso Chacín p. 71-78. ISSN: 8156-4194. <http://www.urbe.edu/publicaciones/telematica/indice/pdf-vol8-1/5-vulnerabilidad-del-protocolo-mysql.pdf> [consulta: 20/11/2011]

SMOOTHWALL (2009). SmoothWall Express 3. [online]. Leeds (UK): SmootWall Ltd. <http://www.smoothwall.org/get/release/3.0.php> [consulta: 20/04/2009]

TAPIA, D. I.; DE PAZ, J. F.; RODRÍGUEZ, S.; ALEGRETE, E. y DE LUIS, A. (2007). Sistema Multi-agente para la Gestión y Monitorización de Rutas de Vigilancia [en línea]. En: BAJO, J.; ALONSO, V.; JOYANES, L. y CORCHADO, J.M. (Eds.) (2007). 6th International Workshop on Practical Applications of Agents and Multiagent Systems (nov.). Salamanca (España): Universidad de Salamanca. p. 269-278, ISBN: 978-84-611-8858-1. <http://bisite.usal.es/webisite/archivos/publicaciones/otrosCongresos/2007/sismulti-agentegestmonitorizarutasvigilancia_b.pdf> [consulta: 18/11/2010]