Diseño de un Sistema de Gestión de Red para Internet de las Cosas

Miguel Durán, *Pontificia Universidad Javeriana, Maestría de Ingeniería Electrónica*

***Abstract*—El presente documento presenta el diseño de un sistema de gestión de red bajo el modelo de computación en el borde para una red de IoT. Este diseño está basado en modelo de gestión de Internet bajo el protocolo SNMP, para la comunicación entre los dispositivos administrados y una Gateway, el cual se comunica con una aplicación de gestión mediante el uso del protocolo SOAP. El sistema permite realizar el monitoreo y configuración de los dispositivos IoT de manera centralizada a través de un único punto que puede estar ubicado en la nube.**

Palabras clave—SNMP, SOAP, MIB, OID, Agente, Gestor, IoT

I. INTRODUCCIÓN

EN la última década, Internet ha evolucionado de ser un repositorio estático de documentos hipertextuales interconectados, a un universo dinámico de personas, aplicaciones y máquinas en red [1]. Actualmente Internet atraviesa por una nueva era conocida como el "Internet de las Cosas" -IoT-, que, de acuerdo con la Unión Internacional de Telecomunicaciones [2], se define como una "*infraestructura mundial para la sociedad de la información que propicia la prestación de servicios avanzados mediante la interconexión de objeto (físicos y virtuales) gracias a la interoperatividad de tecnologías de la información y las comunicaciones presentes y futuras*".

De acuerdo con Ericsson [3], en 2020 serán utilizados 22,3 mil millones de objetos conectados. Para este mismo año, Cisco [4] estima los datos producidos por las personas, las máquinas y las aplicaciones alcanzará los 600 zeta bytes por año. Sin embargo, el tráfico global de los centros de datos solo alcanzará los 15,3 zeta bytes para esa misma fecha.

Debido a la gran cantidad de dispositivos conectados a la red, IoT supone un gran reto en términos de gestión de red, sumado a que muchas veces estos dispositivos no cuentan con capacidad para el procesamiento y análisis de los datos que generan. Adicionalmente, muchas aplicaciones de IoT requieren de tiempos de respuesta bajos y otras requieren de altas tasas de transmisión dada la gran cantidad de datos que producen.

Por lo anterior, ha surgido un nuevo paradigma en IoT denominado Fog Computing, Edge Computing, o en español, computación en el borde, la cual es un escenario donde un gran número de dispositivos heterogéneos (inalámbricos y a veces autónomos), ubicuos y descentralizados se comunican y potencialmente cooperan entre ellos y con la red, para realizar tareas de almacenamiento y procesamiento sin la intervención de la nube

Sin embargo, a pesar de que existe mucho interés en el modelo de computación en el borde, aún existen retos que deben ser resueltos, los cuales están relacionados con la sincronización de las aplicaciones de los dispositivos IoT, la limitación de procesamiento/almacenamiento, la gestión de la red IoT, seguridad, estandarización, monetización y programabilidad de las aplicaciones.

El reto en términos de gestión de red está relacionado con que en el corto plazo va a existir una gran cantidad de dispositivos IoT conectados, muchos de los cuales tienen capacidades limitadas de procesamiento, de almacenamiento y de conexión. Estos dispositivos van a requerir ser configurados y administrados una vez sean desplegados, con el fin de que las aplicaciones que se ejecutan en ellos estén correctamente configuradas y actualizadas para funcionar de forma adecuada. También es importante tener en cuenta que los dispositivos pueden ser características heterogéneas, lo que hace más complejo su administración, pues pueden requerir de configuración específicas.

Teniendo en cuenta lo anterior, en el presente documento se presenta el diseño de un sistema de gestión de red bajo el modelo de computación en el borde para una red de IoT, que permita realizar su monitoreo y configuración de manera eficiente.

II. ESPECIFICACIÓN Y DISEÑO DE LA ARQUITECTURA DEL SISTEMA DE GESTIÓN DE RED

El objetivo del trabajo desarrollado consistió en diseñar un sistema de gestión de red, bajo el modelo de computación en el borde, para una red de IoT, que permita realizar su monitoreo y configuración, e implementar un prototipo que permita validar el diseño realizado. Con base en lo anterior, a continuación, se presentan las condiciones del diseño de arquitectura y funciones de gestión.

A. Diseño de la arquitectura y funciones de gestión

El sistema de gestión de red fue diseñado teniendo en cuenta los tres componentes del modelo de computación en el borde, es decir, los dispositivos IoT, el Gateway IoT y la nube), y bajo una arquitectura estándar de gestión de red (ver Fig. 1). Teniendo en cuenta lo anterior, el desarrollo del sistema de gestión de red implica la definición aspectos relativos a la

información de los dispositivos IoT a ser gestionada, específicamente los atributos o propiedades que permitan realizar su monitoreo y configuración.

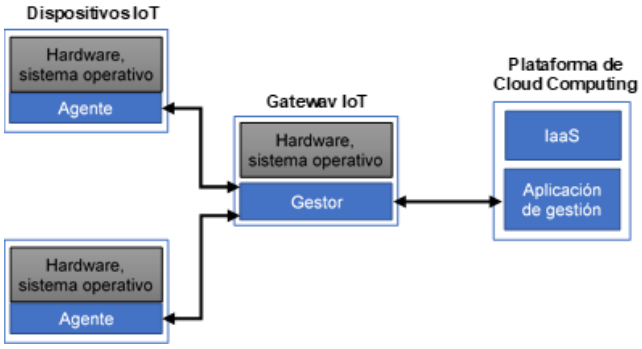

Fig. 1. Componentes del sistema de gestión de dispositivos IoT.

Teniendo en cuenta que se debía seleccionar una arquitectura estándar de gestión de red, fueron evaluadas las 3 principales: i) modelo de gestión OSI; ii) modelo de gestión UIT; y iii) modelo de gestión en Internet. En este sentido, en primer lugar, es importante tener en cuenta que el protocolo de comunicación utilizado en los modelos de OSI y UIT es el CMIP, y el utilizado en el modelo de Internet es el SNMP. Por lo anterior, la selección se basó principalmente en la escogencia del protocolo de comunicación, y la estructura de información asociada a cada uno.

El análisis de estas alternativas dio como resultado la selección del modelo de gestión de Internet para implementar el sistema de gestión de redes IoT, teniendo en cuenta la simplicidad y la baja utilización de memoria y recursos del protocolo SNMP, en comparación con el protocolo CMIP utilizado en los modelos de gestión de OSI y UIT, lo cual es una de las características principales de los dispositivos IoT. Así mismo, a pesar de que de forma nativa SNMP no soporta la realización de tareas, existen implementaciones que permiten ejecutarlas, basadas en la definición de objetos de información desarrolladas para esta necesidad.

TABLA I
COMPARACIÓN DE ATRIBUTOS ENTRE SNMP Y CMIP

Atributo	SNMP	CMIP
Simplicidad de implementación	+	−
Consumo de recursos	−	+
Realización de tareas	+ (no de forma nativa)	+
Transferencia de archivos	+ (no de forma nativa)	+

Por otra parte, en cuanto a la versión de SNMP, se seleccionó la versión 2c, teniendo en cuenta que es la más ampliamente implementada, y que sus funcionalidades permiten la implementación del sistema de gestión para una red IoT. No obstante, en versiones futuras es posible considerar la implementación de la versión 3, la cual entrega mejores opciones de seguridad al protocolo SNMP

El modelo de gestión en Internet especifica una estructura de la información, en la cual se utilizan OID y las bases de información de gestión MIB, en la cual existen unas de tipo estándar. No obstante, dada las características de los dispositivos IoT y de las aplicaciones que soportan, basadas principalmente en el uso de sensores, se realizó el diseño de una MIB de tipo privado, con información específica de dichos dispositivos, que no se encuentra incluida en las MIBs de tipo estándar. La función de esta MIB es entregar información relacionada con el estado de diversos sensores que posee el dispositivo IoT. Con base en lo anterior, la MIB privada para IoT tiene la estructura que se observa en la Fig. 2.

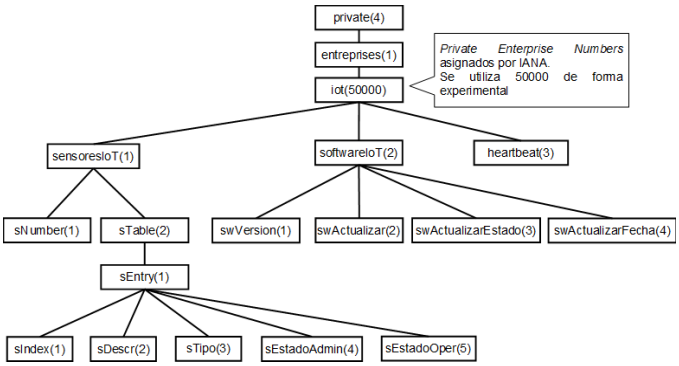

Fig. 2. Estructura de la MIB privada para el sistema de gestión de dispositivos IoT.

En la TABLA II se muestra la descripción de los objetos del subárbol de sensores IoT, el cual está basado en el grupo interfaces de la MIB-II.

Así mismo, en la TABLA III se muestra la descripción de los objetos del subárbol de software IoT, utilizada para la actualización de los dispositivos IoT, y que está basado en la MIB de Cisco OLD-CISCO-FLASH-MIB.

TABLA II
OBJETOS DE SUBÁRBOL DE SENSORES

Objeto	Descripción	Syntaxis	Permiso
sNumber	Número de sensores del dispositivo IoT	INTEGER	read-only
sTable	Lista de sensores presentes en el dispositivo	SEQUENCE of sEntry	not-accessible
sEntry	Entrada que contiene información de un sensor	sEntry	not-accessible
sIndex	Valor único para cada sensor	INTEGER	read-only
sDescr	Cadena de texto que contiene información del sensor	DisplayString	read-only
sTipo	Tipo de sensor	INTEGER { Lista de tipos de sensores }	read-only
sEstadoAdmin	Estado configurado de operación del sensor	INTEGER { up(1), down(2) }	read-write
sEstadoOper	Estado actual de operación del sensor	INTEGER { up(1), down(2) }	read-only

Adicionalmente, se incluyen otros objetos de información, relacionados con otras funciones, o cual se observa en la TABLA IV.

Finalmente, para la gestión de fallas en los dispositivos IoT, se incluyen los traps mostrador en la TABLA V.

TABLA III
OBJETOS DEL SUBÁRBOL DE SOFTWARE IOT

Objeto	Descripción	Sintaxis	Permiso
swVersion	Versión actual de software del dispositivo IoT.	DisplayString	read-only
swActualizar	Nombre de la versión de software a descargar del servidor de archivos	DisplayString	write-only
swActualizar Estado	Estado de la actualización actual o de la última	INTEGER { enProgreso(1), exitoso(2), fallo(3), noAfterPowerOn(4) }	read-only
swActualizar Fecha	Fecha de la última actualización	DisplayString	read-only

TABLA IV
OTROS OBJETOS DE INFORMACIÓN

Objeto	Descripción	Sintaxis	Permiso
heartbeat	Valor en segundos de cada cuanto el dispositivo IoT envía un heartbeat al Gateway IoT	INTEGER	read-only

TABLA V
TRAPS DISEÑADAS

Objeto	Descripción	Sintaxis
heartbeat	Indica que el agente está disponible. El tiempo de envío de este trap está configurado en el OID heartbeat (1.3.6.1.4.1.50000.3.1)	1.3.6.1.6.3.1.1.5.50000.1
sensorDown Admin	Indica que el agente ha detectado que el sensor ha cambiado su estado administrativo a DESACTIVADO. Mediante el valor de este trap se indica el sensor afectado.	1.3.6.1.6.3.1.1.5.50000.2
sensorUp Admin	Indica que el agente ha detectado que el sensor ha cambiado su estado administrativo a ACTIVADO. Mediante el valor de este se trap indica el sensor afectado.	1.3.6.1.6.3.1.1.5.50000.3
sensorDown Oper	Indica que el agente ha detectado que el sensor ha cambiado su estado operativo a DESACTIVADO. Mediante el valor de este se trap indica el sensor afectado.	1.3.6.1.6.3.1.1.5.50000.4
sensorUp Oper	Indica que el agente ha detectado que el sensor ha cambiado su estado operativo a ACTIVADO. Mediante el valor de este trap se indica el sensor afectado.	1.3.6.1.6.3.1.1.5.50000.5
actualizacion	Indica que el dispositivo IoT ha terminado la actualización del software. Mediante el valor de este trap se indica el resultado de este proceso, de acuerdo con los indicados en el OID swActualizarEstado (1.3.6.1.4.1.50000.2.3), el cual puede ser: exitoso(2) o fallo(3). El resultado también puede ser consultado a través del mencionado OID.	1.3.6.1.6.3.1.1.5.50000.6

La comunicación entre el Gateway IoT y la aplicación de gestión debe permitir no solo el intercambio de datos entre los dos nodos, sino que también el intercambio de archivos, con el fin de enviar a través de este medio el software a ser actualizado en los dispositivos IoT.

En materia de IoT, existen múltiples protocolos que permiten realizar el intercambio de datos entre los dispositivos y la nube. En la TABLA VI se muestra una comparación de las opciones analizadas, a partir de lo cual se seleccionó SOAP, principalmente por las características de seguridad que ofrece.

TABLA VI
COMPARACIÓN DE OPCIONES DE COMUNICACIÓN ENTRE EL GATEWAY IOT Y LA APLICACIÓN DE GESTIÓN

Atributo	MQTT	SOAP	REST
Permite intercambio de datos	Si	Si	Si
Permite intercambio de archivos	No	Si	Si
Consumo de recursos	Bajo	Alto	Medio
Medidas de seguridad	Medio (SSL/TLS)	Alto (WS-Security y SSL/TLS)	Medio (SSL/TLS)

B. Diseño del agente del dispositivo IoT

Teniendo en cuenta las funciones de gestión en el dispositivo IoT, el diseño desarrollado para el agente está conformado por 4 módulos, tal y como puede observarse en la Fig. 3.

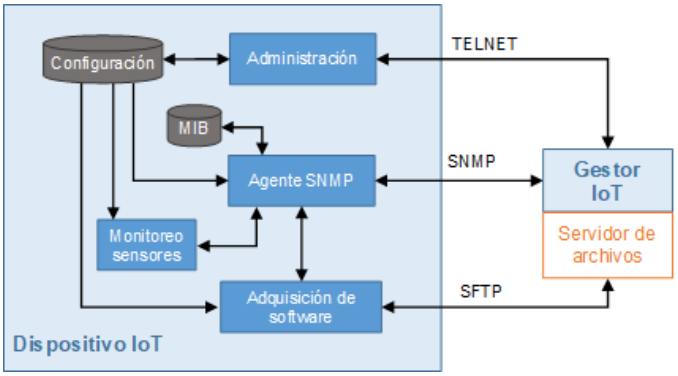

Fig. 3. Componentes del agente del dispositivo IoT

El módulo “agente SNMP” implementa las funcionalidades del protocolo SNMP, desde el punto de vista del agente, que como se indicó previamente, corresponde a la versión 2c, teniendo en cuenta que posteriormente puede ser implementada la versión 3.

El módulo de “monitoreo de sensores” se encarga de monitorear el estado de sensores, es decir, si están activados o desactivados, ya sea como consecuencia de procesos de administración o por operación.

El módulo de “adquisición de software” se encarga de realizar la actualización del software del dispositivo IoT, asociado a la aplicación de IoT que se está ejecutando. Es importante tener en cuenta que esto no incluye actualización del firmware de dispositivo IoT o de su sistema operativo.

Finalmente, el módulo de “administración” permite realizar

la configuración vía remota del dispositivo IoT, a través del acceso y modificación a la base de datos de configuración.

C. Diseño del gestor del Gateway IoT

Teniendo en cuenta las funciones de gestión en el Gateway IoT, el gestor está conformado por 5 módulos, tal y como puede observarse en la Fig. 4.

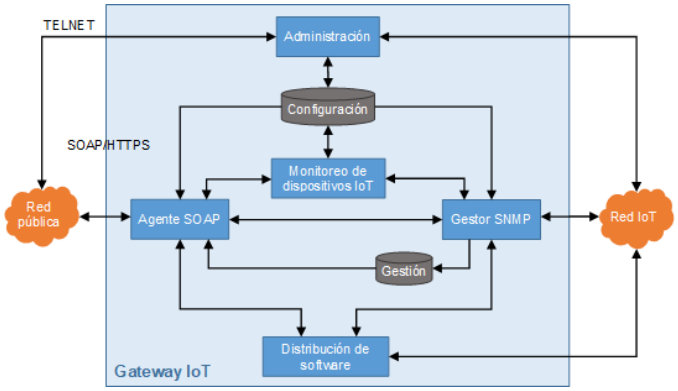

Fig. 4. Componentes del gestor del Gateway IoT

El módulo "gestor SNMP" implementa las funcionalidades del protocolo SNMP, desde el punto de vista del gestor, que como se indicó previamente, corresponde a la versión 2c, teniendo en cuenta que posteriormente puede ser implementada la versión 3.

El módulo de "monitoreo de dispositivos IoT" se encarga de monitorear el estado de los dispositivos IoT, es decir, si están disponibles o no.

El módulo "agente SOAP" se encarga de realizar la comunicación entre el Gateway IoT y la aplicación de gestión, a través del protocolo SOAP.

El módulo de "distribución de software" se encarga de realizar la distribución del software de actualización a los dispositivos IoT.

Finalmente, el módulo de "administración" permite realizar la configuración vía remota del Gateway IoT, a través del acceso y modificación a la base de datos de configuración.

D. Diseño de la aplicación de gestión de la plataforma de computación en la nube

Teniendo en cuenta las funciones de gestión en la aplicación de gestión, esta está conformada por 4 módulos, tal y como puede observarse en la Fig. 5.

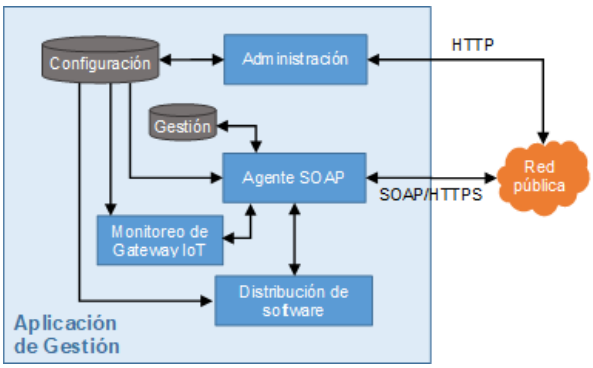

Fig. 5. Componentes de la aplicación de gestión en la nube

El módulo "agente SOAP" se encarga de realizar la comunicación entre la aplicación de gestión y el Gateway IoT, a través del protocolo SOAP.

El módulo de "monitoreo de Gateway IoT" se encarga de monitorear el estado de los Gateway IoT.

El módulo de "distribución de software" se encarga de realizar la distribución del software de actualización a los dispositivos IoT. Para lo anterior, realiza la interacción con módulo "Agente SOAP", para el envío del software de actualización a los Gateway IoT

Finalmente, el módulo de "administración" permite realizar la configuración vía remota de parámetros de la aplicación de gestión de la nube, la cual es almacenada en una base de datos de configuración. Así mismo, a través de este módulo se realiza la visualización y modificación de la información de los dispositivos IoT enviada por los Gateway IoT, a través de una interfaz Web (ver Fig. 6).

Fig. 6. Interfaz web de la aplicación de gestión

III. CONCLUSIONES

El diseño de la arquitectura del sistema de gestión de red desarrollado permite administrar de manera simple los dispositivos desplegados en una red IoT. La estructura del modelo de objetos de información diseñado permite gestionar de manera sencilla la información de los sensores de los dispositivos IoT.

Por otra parte, la heterogeneidad de las características de los dispositivos y sensores existentes en el mercado, pueden ser homogeneizadas a través del sistema de gestión de red diseñado. En este sentido, la estructura diseñada permite a futuro agregar características adicionales de los sensores o del dispositivo IoT en sí mismo, en beneficio de esta homogenización.

Los usuarios potenciales del prototipo implementado son aquellas empresas que tengan implementada una red de dispositivos IoT y necesiten una herramienta que permita realizar la administración de estos.

En cuanto al prototipo implementado, este permitió realizar la validación del funcionamiento de las características diseñadas. Teniendo en cuenta que la implementación fue realizada en su mayoría mediante lenguaje java, a futuro se puede evaluar la misma mediante el uso del lenguaje C, específicamente para el dispositivo IoT y el Gateway IoT.

Finalmente, los futuros desarrollos en el sistema de gestión de red diseñado pueden estar relacionados con la implementación de SNMP v3 a nivel de la comunicación entre

el dispositivo IoT y el Gateway IoT. Así mismo, en cuanto a la comunicación entre el Gateway IoT y la aplicación de gestión, a pesar de que el protocolo seleccionado fue SOAP, se podrían implementar protocolos adicionales como REST, permitiéndoles al usuario su selección, dependiente de las características de desempeño que necesite de manera particular.

REFERENCIAS

[1] Jadoul, M., "The IoT: The next step in internet evolution," Nokia, Mar. 15, 2015. [En línea]. Disponible: https://www.nokia.com/blog/iot-next-step-internet-evolution/

[2] Recomendación UIT-T Y.2060. Descripción general de Internet de los objetos, Unión Internacional de las Telecomunicaciones, 2012.

[3] Jejdling, F., "Ericsson Mobility Report," Ericsson, Nov., 2018. [En línea]. Disponible: https://www.ericsson.com/assets/local/mobility-report/documents/2018/ericsson-mobility-report-november-2018.pdf

[4] Evans, D., "Internet de las cosas. Cómo la próxima evolución de Internet lo cambia todo," Cisco Internet Business Solutions Group, Abr., 2011. [En línea]. Disponible: https://www.cisco.com/c/dam/global/es_mx/solutions/executive/assets/pdf/internet-of-things-iot-ibsg.pdf

[5] Yi, S., Hao, Z., Qin, Z. & Li, Q. (2015). Fog Computing: Platform and Applications. Third IEEE Workshop on Hot Topics in Web Systems and Technologies. Computer Science and Information Systems.

[6] Bonomi, F., Milito, R., Natarajan, P. & Zhu, J. (2014). Fog Computing: A Platform for Internet of Things and Analytics. Springer International Publishing.

[7] Bonomi, F., Milito, R., Zhu, J. & Addepalli, S. (2012). Fog Computing and Its Role in the Internet of Things. Cisco Systems Inc.

[8] Dastjerdi, A., Gupta, J., Calheiros, R., Ghosh, S & Buyya, R. (2016). Internet of Things. Principles and Paradigms. Chapter 4. Fog Computing: Principles, Architectures, and Applications. ELSEVIER.

[9] Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge Computing: Vision and Challenges. IEEE Internet of Things Journal, Vol. 3, No. 5.

[10] Martí, A. (1999) Gestión de redes. Universidad de Cataluña.

[11] Recomendación UIT-T X.701. (1997) Gestión de interconexión de sistemas abiertos - Marco y arquitectura de la gestión de sistemas. Unión Internacional de Telecomunicaciones.

[12] Romero, M. Gestión de Redes. Sistemas Avanzados de Comunicaciones. Universidad de Sevilla. [En línea]. Disponible: http://www.dte.us.es/personal/mcromero/docs/sac/sac-gestionderedes.pdf

[13] Recomendación UIT-T M.3010. (2000) Principios para una red de gestión de las telecomunicaciones. Unión Internacional de Telecomunicaciones.

[14] Weber, J. Fundamentals of IoT device management. Avnet. [En línea]. Disponible: http://iotdesign.embedded-computing.com/articles/fundamentals-of-iot-device-management/

[15] Rodríguez, D. (2013). Arquitectura y Gestión de la IoT. Revista Telem@tica Vol. 12, No. 3 (pp. 49-60). Instituto Superior Politécnico José Antonio Echeverría.

[16] Hui-Ping, H., Shi-De, X., & Xiang-Yin, M. (2015). Applying SNMP Technology to Manage the Sensors in Internet of Things. The Open Cybernetics & Systemics Journal (pp. 1019-1024).

[17] Choi, H., Kim, N., & Cha, H. (2009). 6LoWPAN-SNMP: Simple network management protocol for 6LoWPAN. 11th IEEE International Conference on High Performance Computing and Communications (pp. 305-313). IEEE.

[18] Subramanian, M. (2000). Network Management: Principles and Practices. Addison Wesley.

[19] Luque, J., Gonzalo, F., Escudero, J. & Carrasco, A. (1999). TMN versus SNMP-based network management systems. Universidad de Sevilla. [En línea]. Disponible: http://personal.us.es/jluque/Congresos/1999%20Cigre%20Polonia-1.pdf

[20] A guide to understanding SNMP. (2013). SolarWinds Worldwide, LLC.

[21] Mauro, D. & Schmidt, K. (2005) Essential SNMP. Segunda edición. O'Reilly.

[22] Perkins, D. (1993) Understanding SNMP MIBs. Revision 1.1.7.

[23] Router Teldat. Agente SNMP. (2000) Doc. DM512 Rev. 8.40. [En línea]. Disponible: http://www.it.uc3m.es/~teldat/Cbra/castellano/protocolos/Dm512v840_Agente_SNMP.PDF