Gestión de red

Pere Barberán

PID_00201838

UOC
Universitat Oberta de Catalunya

Índice

Introducción 5

Objetivos 6

1. Introducción a la gestión de red 7
1.1. Fundamentos de la gestión de red 8
1.1.1. Requisitos de la gestión de red 8
1.1.2. Componentes de la gestión de red 9

2. Gestión de red SNMP 10
2.1. Orígenes de la gestión de red SNMP 10
2.2. Arquitectura de gestión de red SNMP 11
2.3. La base de datos de información en SNMP 12
2.3.1. Estructura de datos de información. Modelo de información 13
2.3.2. Estructura de la MIB 14
2.3.3. Definición de los objetos 16
2.3.4. Las MIB privadas 17
2.4. El protocolo SNMP 17
2.4.1. Operaciones en SNMP 18
2.4.2. Las comunidades SNMP 18
2.4.3. Identificación de instancias 19
2.4.4. Formato del paquete SNMP 22
2.5. Limitaciones del protocolo SNMP 22

3. SNMPv2 24
3.1. Mejoras en la gestión 24
3.2. Estructura de la información de gestión 24
3.3. El protocolo SNMPv2 25

4. RMON (*remote network monitoring*) 26
4.1. Objetivos del diseño de RMON 26
4.2. Control de la sonda RMON 26
4.2.1. Configuración de la sonda 27
4.3. RMON1 y RMON2 28
4.3.1. La MIB RMON 29
4.3.2. RMON2 30

5. Gestión TMN 32
5.1. Arquitectura TMN 34
5.1.1. Arquitectura funcional del TMN 34
5.1.2. Arquitectura física del TMN 35
5.1.3. Arquitectura de información del TMN 35

Resumen 36

Actividades 37

Ejercicios de autoevaluación 37

Solucionario 38

Glosario 40

Bibliografía 40

Introducción

El imparable crecimiento que se ha producido en los últimos veinticinco años en las necesidades de procesamiento de la información ha ido acompañado del rápido desarrollo de los ordenadores y de la tecnología de red para satisfacer estas necesidades. Junto a esta demanda se ha dado una explosión en la variedad de equipos y redes ofrecidas por los vendedores. Quedan lejos los días en los que las empresas confiaban en un único vendedor y, en consecuencia, en una arquitectura que cubriera sus necesidades.

Hoy en día, las organizaciones crecen rápidamente y su arquitectura de red puede estar formada por diferentes LAN, WAN, soportadas por conmutadores (*switchs*), encaminadores, diferentes tipos de computadores, servicios, etc. Para gestionar estos sistemas y redes son necesarias herramientas y aplicaciones que trabajen de manera automatizada. La base de estas herramientas en un entorno multivendedor es que haya técnicas estandarizadas tanto para representar como para intercambiar la información de gestión de redes.

Así, en el mundo de Internet el estándar elegido ha sido *simple network management protocol* (SNMP) y la especificación relacionada *Remote network monitoring* (RMON). Estos estándares iniciales han sido revisados y actualizados con varias versiones en los últimos años.

Dentro del mundo de las telecomunicaciones el proceso ha sido similar en cuanto a crecimiento. Actualmente resulta muy competitivo, vista la liberalización y globalización del mercado de las telecomunicaciones. Los estándares TMN están creciendo en importancia como un medio para proporcionar una ventaja a los proveedores de servicios de telecomunicaciones.

En este módulo didáctico se quiere dar una visión de los conceptos asociados a la gestión de red y los protocolos más ampliamente aceptados, como son SNMP y TMN.

Objetivos

Los materiales didácticos de este módulo deben permitir que el estudiante alcance los siguientes objetivos:

1. Entender la necesidad de gestión de red, sus requisitos y la arquitectura básica de la gestión.

2. Conocer los elementos de la gestión de red SNMP profundizando en cada uno de estos elementos y viendo las ventajas y las limitaciones de las diferentes versiones del protocolo.

3. Aprender la estructura de la MIB-II y ser capaz de llevar a cabo peticiones SNMP en los dispositivos gestionados.

4. Entender los objetivos de la monitorización remota RMON, cómo se configura y cómo se accede a los recursos almacenados en las sondas.

5. Diferenciar las necesidades de las operadoras de telecomunicaciones con respecto a las redes convencionales

6. Conocer la arquitectura básica de gestión TMN para infraestructuras de telecomunicaciones.

1. Introducción a la gestión de red

Igual que las redes transportan información, esta información necesita ser transmitida de manera eficiente, efectiva y fiable. Existen muchas razones que hacen que sea necesaria una gestión de red:

- Las redes actuales son una colección de hardware y software de fabricantes diferentes que es necesario configurar para que funcionen conjuntamente. Cada vendedor a la vez tiene sus herramientas para gestionar sus equipos de la manera más adecuada posible.
- Las nuevas tecnologías deben convivir con tecnologías antiguas antes de que éstas sean actualizadas. Por lo tanto, es necesario configurar redes con tecnologías de varias "generaciones".
- La gestión de red es un factor de coste. Las empresas intentan que la planificación, el apoyo, la detección, la corrección de problemas, la operación y la administración sean tareas que tengan un componente lo más automatizado posible.
- La seguridad constituye otro aspecto muy importante en las redes actuales.
- Otros aspectos relacionados con la gestión son la planificación y el diseño, la gestión de políticas, la gestión de sistemas y la gestión de servicios que se llevan a cabo con herramientas normalmente propietarias.

Así, el marco de trabajo de la gestión de red debe estar dirigido a resolver estos temas dentro de límites funcionales y de coste. El coste de la gestión de red no se refiere sólo al precio de compra de los equipos, sino que se tiene que ver desde un punto de vista más amplio y debe prever aspectos humanos, actualizaciones, utilización de los sistemas, etc.

Se puede decir, por lo tanto, que la gestión de red optimiza la capacidad funcional de la red. Algunas connotaciones que se pueden derivar de esto son las siguientes:

- Mantiene la red operando a rendimiento máximo.
- Informa al operador de cualquier anomalía e intenta acotar el lugar donde se ha producido.
- Posibilita examinar el rendimiento de la red en tiempo real.
- Proporciona funciones de gestión de la configuración.

El objetivo y los beneficios de las plataformas actuales de gestión de red son los siguientes:

- Mejorar en el proceso de la toma de decisiones.
- Mejorar en la eficacia de las operaciones.
- Liberar a los gestores de los trabajos rutinarios para poder dedicarse al análisis, el control y la planificación. Gran parte de los fallos suelen ser rutinarios.
- Alcanzar el rápido retorno de la inversión.

1.1. Fundamentos de la gestión de red

¿Qué es la gestión de red?

> La ISO define la gestión de red como el conjunto de elementos de control y supervisión de los recursos que permiten que la comunicación tenga lugar sobre la red.

1.1.1. Requisitos de la gestión de red

La ISO ha definido las principales áreas funcionales de la gestión de red. Aunque esta clasificación ha sido desarrollada por el OSI, ha obtenido un amplio reconocimiento. Las áreas funcionales son las siguientes:

1) Gestión de fallos. Cuando se produce un fallo es muy importante que el administrador pueda actuar lo más rápidamente posible:

a) Determinar dónde se ha producido el fallo.
b) Aislar el resto de la red del fallo para que pueda continuar funcionando sin interferencias.
c) Reconfigurar o modificar la red de manera que se minimice el impacto de operación sin que fallen los diferentes componentes.
d) Reparar o reemplazar el componente para que la red vuelva a funcionar de manera correcta.

2) Gestión de rendimiento. Se define como la evaluación del comportamiento de los elementos de la red. Para poder realizar este análisis es necesario mantener uno histórico con datos estadísticos y de configuración.

3) Gestión de cuentas. Determinación de los costes asociados al uso de los recursos y la asignación de las cargas correspondientes.

4) Gestión de configuración. La gestión de configuración se encarga de las siguientes tareas:

a) Inicialización y desactivación.
b) Definición o cambio de parámetros de configuración.
c) Recogida de información de estado.
d) Denominación de los elementos de la red.

5) Gestión de seguridad. Incluye toda una serie de facilidades mediante las cuales el administrador de red modifica las funciones que proporcionan seguridad ante los intentos de accesos no autorizados. Encontraríamos aspectos como la gestión de claves, los cortafuegos o los históricos de seguridad.

Alcance de los requisitos de la gestión de la red

En la mayoría de casos las plataformas actuales no cubren todos los requisitos de la gestión de la red.

1.1.2. Componentes de la gestión de red

La gestión de red está formada básicamente por cuatro componentes (figura 1):

1) Plataforma de gestión o elemento que supervisa. Este componente se denomina *network management server* (NMS) y se usa para gestionar toda la red. Recibe toda la información y la muestra una vez que **filtra** lo que realmente es importante.

Función del gestor de la red

Si el gestor no hiciera un filtrado de la información que realmente es útil, la gestión de red acabaría no resultando interesante debido al exceso de información. Es necesario, por tanto, que el gestor elimine la información superflua.

2) El elemento para gestionar. Este elemento se denomina *network management agent* (NMA). El NMA es un elemento de la red como pueden ser los encaminadores, commutadores, *hosts*, etc. A la vez, cada nodo de red contiene una colección de software encargado de las tareas de gestión. Este software se denomina NME (*network management entity*).

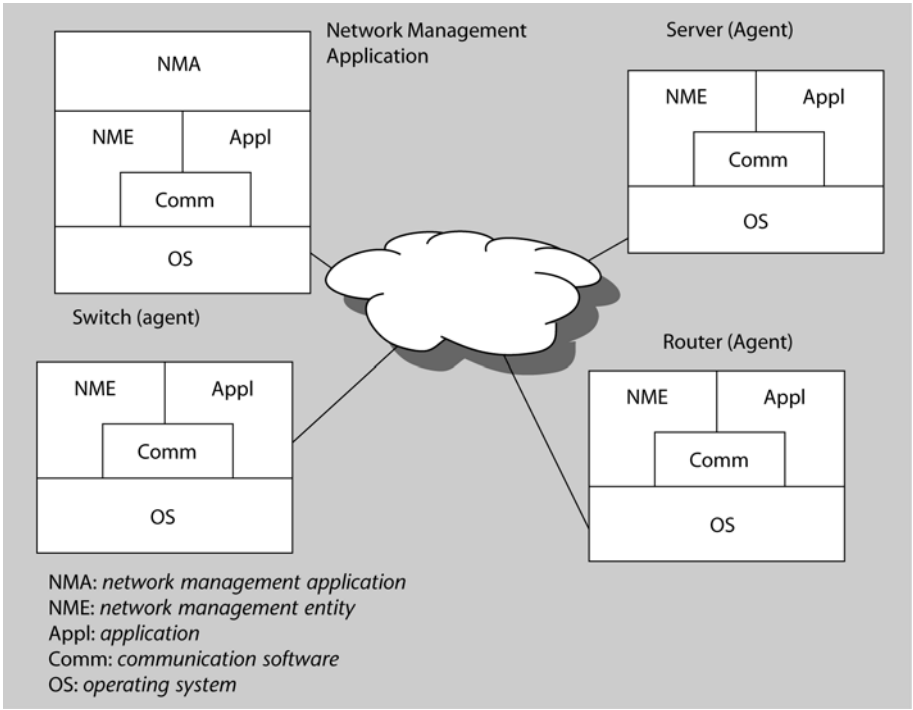
3) El protocolo de diálogo entre el agente y el servidor. El protocolo más común es SNMP (*simple network management protocol*). Genéricamente sería el software de comunicación.

4) Los objetos para gestionar. Estos objetos son los recursos de los dispositivos para gestionar. Se agrupan normalmente en una especie de base de datos denominada MIB (*management information base*).

Ejemplos de objetos para gestionar

Algunos ejemplos de objetos para gestionar pueden ser las tablas de encaminamiento de un encaminador, la última vez que se reinició un equipo, etc.

Figura 1. Elementos de un sistema de gestión



2. Gestión de red SNMP

La gestión de red SNMP (*simple network management protocol*) engloba todo un grupo de especificaciones de gestión de red que incluyen el **protocolo**, la **definición de la estructura de datos** y otros conceptos asociados.

2.1. Orígenes de la gestión de red SNMP

En los inicios de lo que conocemos como Internet no existían los protocolos de gestión como tal. La única herramienta que se utilizaba para "gestión" era el protocolo ICMP (*Internet control message protocol*). ICMP se encuentra disponible en todos los dispositivos IP y en sus inicios era un mecanismo que servía para testear (*echo/reply*) la comunicación entre parejas de dispositivos. El receptor del mensaje *echo* está obligado a responder a quien le realiza la petición. A la vez, también resulta útil para obtener otras informaciones muy interesantes, como el retraso en la red.

El ejemplo más conocido del uso del protocolo ICMP es el programa **ping** (*packet Internet grouper*). Con esta sencilla herramienta se pueden observar, por ejemplo, las variaciones en el retraso extremo a extremo, la pérdida de paquetes, de manera que puede ayudar entre otras cosas a determinar áreas de congestión y puntos de fallo.

Figura 2. Captura de una petición ping desde *cmd*

```
Z:\>ping 147.83.245.4

Haciendo ping a 147.83.245.4 con 32 bytes de datos:

Respuesta desde 147.83.245.4: bytes=32 tiempo=1ms TTL=255
Respuesta desde 147.83.245.4: bytes=32 tiempo=1ms TTL=255
Respuesta desde 147.83.245.4: bytes=32 tiempo=44ms TTL=255
Respuesta desde 147.83.245.4: bytes=32 tiempo=1ms TTL=255

Estadísticas de ping para 147.83.245.4:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 1ms, Máximo = 44ms, Media = 11ms

Z:\>
```

Esta herramienta junto con otras herramientas sencillas fueron soluciones lo suficientemente válidas para la gestión de red hasta mediados de los años ochenta. A partir de entonces, con el crecimiento exponencial de Internet se hizo necesario plantear nuevas herramientas para la gestión.

Las propuestas más interesantes fueron:

- *Simple network management protocol* (SNMP)
- CMIP sobre TCP/IP (CMOT)

En 1988, Internet Architecture Board (IAB) revisó las propuestas y aprobó que SNMP fuera desarrollado como una solución a corto plazo, mientras que se planteaba CMOT como la solución hacia donde se debía ir. La simplicidad del protocolo SNMP hizo que pronto fuera el protocolo elegido por los fabricantes.

Al realizar una breve cronología de la evolución de SNMP encontramos lo siguiente:

1) **SNMPv1**. Surge e introduce las bases en la gestión de redes.

2) **SNMPv2**. Mejora capacidades de transferencia de datos e intercambio de información entre gestores (*getbulkrequest*, *informrequest*) y mejora la manera de gestionar las tablas.

3) **RMONv1**. Permite la monitorización remota de un segmento de red.

4) **RMONv2**. Amplía la MIB-II de RMON y permite una monitorización de segmentos de redes LAN y un seguimiento del tránsito en red y su aplicación.

5) **SNMPv3**. Introduce capacidades de seguridad (control de acceso, autenticación y privacidad).

2.2. Arquitectura de gestión de red SNMP

El modelo de gestión de red TCP/IP tiene los elementos clave siguientes:

- La estación de gestión o gestor
- Los agentes gestionados
- La base de datos de gestión (*managament information base*, MIB)
- El protocolo de gestión de red

La **estación de gestión** puede ser uno o un grupo de dispositivos distribuidos. La estación de gestión sirve de interfaz entre el gestor y el sistema de gestión de red.

Si os fijáis en ello, los elementos de la red TCP/IP son los mismos ya comentados en el subapartado 1.1.2.

Los **agentes gestionados** son el equipamiento lógico alojado en los dispositivos para gestionar de la red. Almacena datos de gestión y responde a las peticiones asociadas a estos datos. De manera asíncrona (*traps*) –de los cuales hablaremos más adelante– pueden proporcionar información al gestor sin que éste lo solicite.

La **base de datos de gestión (*managament information base*, MIB)** es una base de datos virtual de los recursos por gestionar, los cuales se representan como ob-

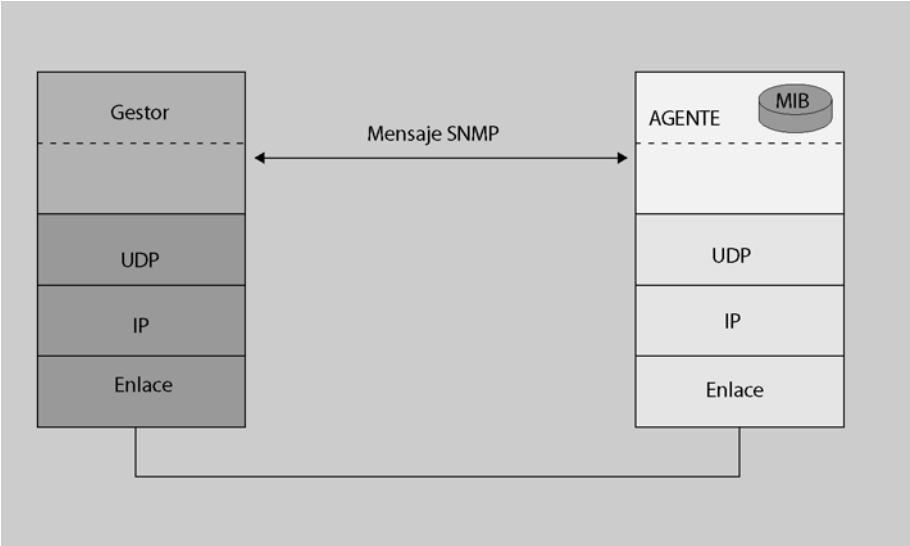
En el subapartado 2.3 trataremos con detalle la MIB y sus objetos.

jetos. Estos objetos de la MIB están **estandarizados**, son accesibles por el agente y pueden ser manipulados vía SNMP para realizar la gestión de red.

Finalmente, el último elemento básico dentro de la arquitectura es el protocolo de comunicación, que permite comunicar la estación de gestión con los agentes. En este caso es el protocolo SNMP. Es un protocolo que se basa en el paradigma **petición-respuesta**.

En la figura 3 tenemos el esquema básico con los cuatro elementos descritos.

Figura 3. Modelo de arquitectura de gestión SNMP



Protocolo SNMP

El protocolo SNMP está montado sobre UDP, que es un protocolo de capa de transporte no orientado a conexión.

2.3. La base de datos de información en SNMP

La base de datos de información se llama MIB (*management information base*) y contiene la información de los elementos para ser gestionados.

La MIB está estructurada de manera **jerárquica**, donde cada recurso está representado por un objeto.

En el subapartado 2.3.2 podéis ver cómo está formada la estructura jerárquica de la MIB.

La MIB debe cumplir un par de reglas básicas:

1) El objeto u objetos utilizados para representar un recurso particular ha de ser el mismo para todos los sistemas.

Sería imposible crear un protocolo para adquirir información si esta información dependiera del sistema por gestionar.

2) Ha de existir un esquema común de representación para soportar interoperabilidad. Este esquema común es la **estructura de datos de información (SMI)**.

2.3.1. Estructura de datos de información. Modelo de información

El SMI define el tipo de datos que se pueden utilizar y especifica cómo se pueden representar y nombrar los recursos.

SNMP utiliza como notación un subconjunto de *abstract syntax notation one* (ASN.1) para acomodar la comunicación entre sistemas heterogéneos.

ASN.1 es un formato de datos que es común a todas las máquinas y, por lo tanto, es independiente del hardware o sistema operativo. Para hacer esta representación de los datos se podía haber elegido algún lenguaje de alto nivel ya existente, como C, pero lo que se ha querido es que sea independiente del lenguaje.

Implementación del ASN.1

El ASN.1 es una notación común para representar los datos, pero su implementación se puede hacer en el lenguaje que se quiera.

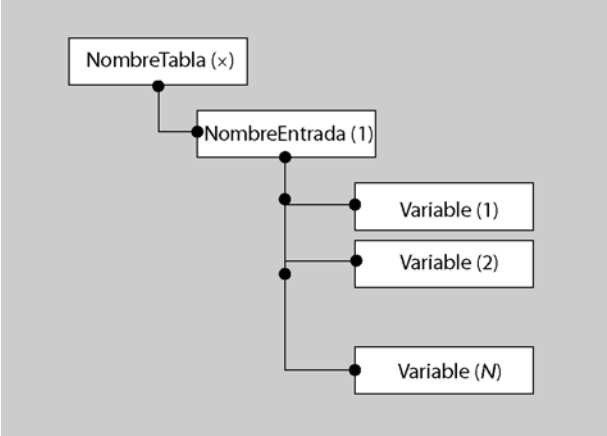
La MIB SNMP almacena, de todos los tipos definidos por el ASN.1, sólo los tipos de datos simples, entendiendo por simples lo siguiente:

Escalares. Así, las posibles variables son las que se muestran en la tabla siguiente.

Nombre	Tipo	Bytes	Significado
INTEGER	Numeric	4	Enter 32 bits
Counter32	Numeric	4	Unsigned 32 bits counter that wraps
Gauge32	Numeric	4	Unsigned value that does no wrap
Integer32	Numeric	4	32 bits
Unteger32	Numeric	4	Like integer32 but unsigned
Counter64	Numeric	8	En 64 bit counter
TimeTicks	Numeric	4	
BIT STRING	String	4	Bit map of 1 to 32 bits
OCTET STRING	String	= 0	Variable length byte string
Opaque	String	= 0	Obsolete
OBJECT IDENTIFIER	String	> 0	A list of integers
ipAddress	String	4	A dotted decimal integer address
NsapAddress	String	< 22	An OSI NSAP address

***Arrays* bidimensionales de escalares (tablas)**. La manera de definir una tabla, que es usada por todas las tablas de la MIB, es mediante dos tipos ASN.1, que son: SEQUENCE y SEQUENCE OF. SEQUENCE es comparable a lo que entendemos por un *array en* programación, mientras que SEQUENCE OF es comparable a una estructura. De esta manera, la tabla se crea a partir de un *array* donde cada elemento del *array* es una estructura de escalares. La forma se muestra en la figura 4.

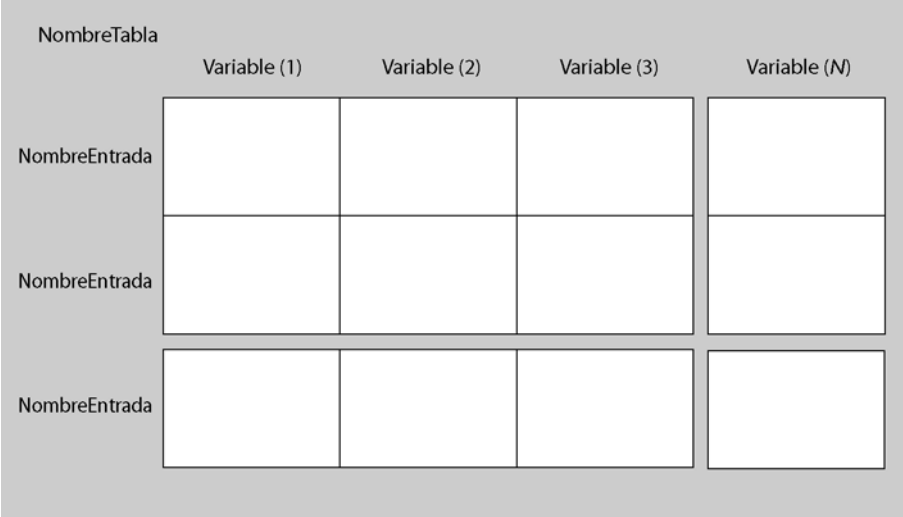
Figura 4. Ejemplo de tabla genérica SNMP



Las filas de la tabla estarán formadas por elementos *NombreEntrada* los cuales constituyen una SEQUENCE. Al mismo tiempo, una de las variables será los índices. Éstos índices permitirán distinguir cada una de las filas dentro de la tabla (figura 5).

Más adelante hablaremos del uso de los indices.

Figura 5. Representación de una tabla SNMP



2.3.2. Estructura de la MIB

Para definir la estructura de la MIB lo primero que se ha de hacer es tener claro cómo se quieren definir los objetos. Así, la estructura utilizada sigue las siguientes reglas básicas:

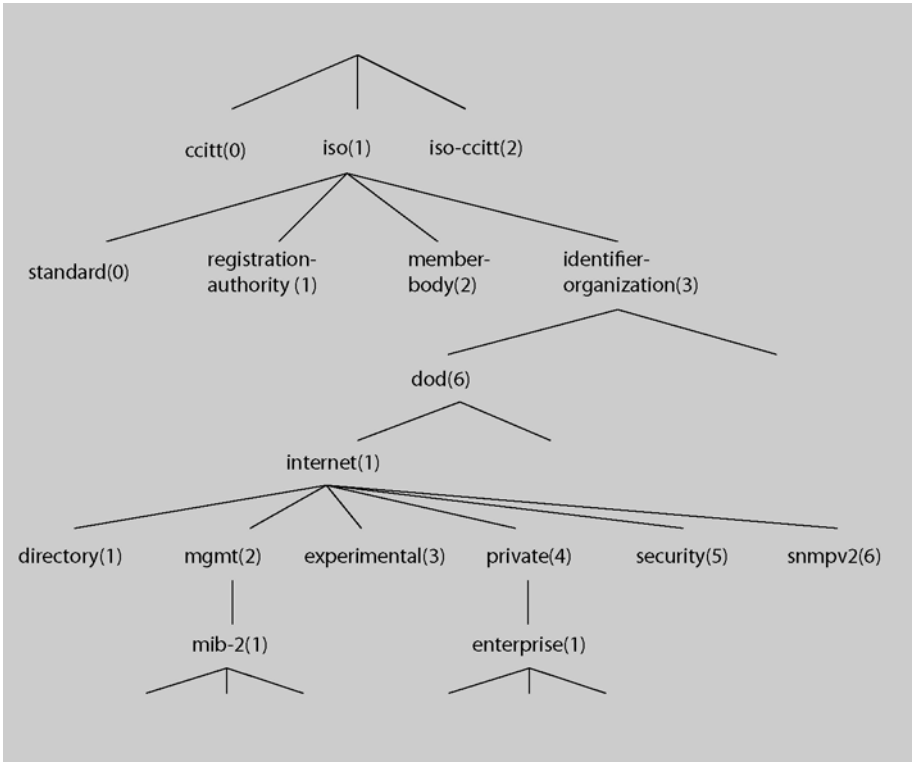
- Identificación no ambigua y universal de los objetos mediante una arquitectura en forma de árbol.

- Esta estructura está basada en el esquema de identificación de objetos definido por el OSI.

Siguiendo este esquema del OSI los objetos gestionados en el entorno SNMP están ordenados en una arquitectura de árbol donde las ramas son los diferen-

tes objetos. Asociado a cada tipo de objeto hay un identificador de tipo ASN.1 (*object identifier*) que sirve para nombrar los objetos y para situarse dentro de la estructura del árbol. La jerarquía MIB se puede ver como un árbol con una raíz sin nombre y donde tenemos varios niveles, asignados por diferentes organizaciones (figura 6).

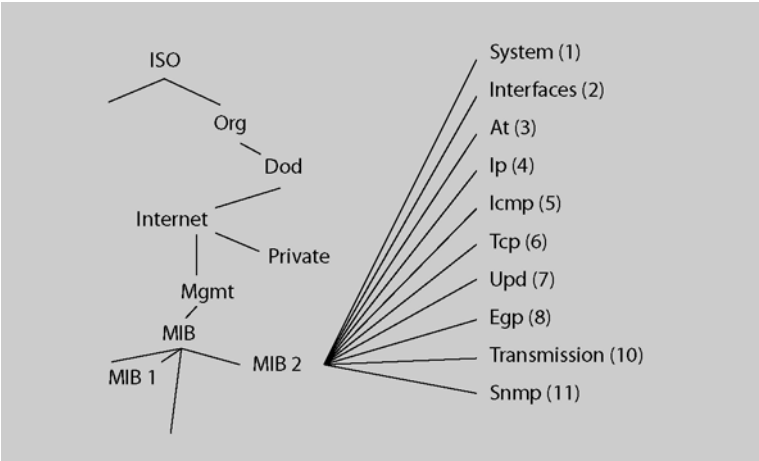
Figura 6. Representación de forma jerárquica de la MIB



Un aspecto muy importante en la comunicación entre dispositivos heterogéneos es que exista una representación de los datos que sea común. Así, esta tarea, como ya se ha dicho, la realiza la notación estandarizada ASN.1.

Dentro de la MIB-II encontramos los grupos que se muestran en la figura 7.

Figura 7. Grupos de la MIB-II dentro de la estructura jerárquica



En la tabla siguiente se muestran los diez grupos administrados con una breve descripción de cada uno de ellos.

Grupo	Objetos	Descripción
System	7	Nombre, localización y descripción del equipo
Interfaces	23	Interfaces de red y estadísticas de tráfico
At	3	Translación de direcciones (no usado)
Ip	42	Estadísticas de los paquetes IP
Icmp	26	Estadísticas de paquetes ICMP recibidos
Tcp	19	Algoritmos, parámetros y estadísticas de tráfico TCP
Udp	6	Estadísticas de tráfico UDP
Egp	20	Estadísticas de tráfico EGP
Transmission	0	Reservado
snmp	29	Estadísticas de tráfico SNMP

2.3.3. Definición de los objetos

Una MIB está formada por objetos. Cada objeto definido tiene asociado un tipo y un valor. Así, como podéis imaginar, existen muchos tipos diferentes en función de la información de los objetos.

Por sencillez se usa una macro para crear los diferentes objetos. Esta macro define la sintaxis para los diferentes conjuntos de tipos relacionados.

```
OBJECT-TYPE MACRO ::=
BEGIN
  TYPE NOTATION ::= "SYNTAX" type (TYPE ObjectSyntax )
                  "ACCESS" Access
                  "STATUS" Status
  VALUE NOTATION ::= value (VALUE ObjectName)
  Access ::= "read-only"
           | "read-write"
           | "write_only"
           | "non-accessible"
  Status ::= "mandatory"
           | "optional"
           | "obsolete"
END
```

A partir de ésta se crean los tipos específicos, lo que denominamos "instancias macro".

```
OBJECT:
  object descriptor object identifier
SYNTAX:
  sintaxis ASN.1 de los objetos
DEFINITION:
  descripción del objeto
ACCESS:
  sólo lectura, lectura-escritura o no accesible
STATUS:
  obligatorio, opcional u obsoleto
```

Ejemplo de definición de objeto

A continuación tenéis un ejemplo de definición de objeto que proporciona el RFC 1212

```
ifNumber OBJECT-TYPE
 SYNTAX INTEGER
 ACCESS read-only
 STATUS mandatory
 DESCRIPTION
 "The number of network interfaces (regardless of
 their current state) present on this system."
 ::= { interfaces 1 }
```

En el caso del ejemplo, cuando se quiere nombrar este objeto concreto es necesario hacer: *Iso (1) identified-organization (3) dod (6) internet (1) mgmt(2) mib-2(1) interfaces (2) IfNumber (1)* o la forma como se expresa habitualmente 1.3.6.1.2.1.2.1.

Para poder pedir la información concreta de la variable asociada a este objeto es necesario incluir un 0: 1.3.6.1.2.1.2.1.0

Web recomendada

Los RFC (*request for comments*) contienen información técnica y documentos asociados a Internet que incluyen especificaciones técnicas y documentos políticos producidos por Internet Engineering Task Force (IETF). Los podéis consultar en http://www.rfc-editor.org.

En el subapartado 2.4.3 podéis ver en detalle cómo se realiza la petición a objetos concretos (instancias).

Un tipo de objeto que es necesario tratar aparte lo constituyen las tablas. SMI soporta sólo una forma de estructura de datos: **tablas bidimensionales de escalares**.

2.3.4. Las MIB privadas

Una de las características más importantes de SNMP es el hecho de que la MIB ha sido diseñada de manera que puede ir creciendo y así proporciona flexibilidad para incorporar nuevos objetos. Si os fijáis en la figura 7, donde se muestra la estructura MIB hay una rama *private* a la que se pueden añadir extensiones. Esto permite a los fabricantes incorporar objetos asociados en la gestión específica de sus productos. Gracias a la estandarización del SMI, aunque creemos objetos privados éstos se pueden gestionar desde cualquier plataforma de gestión, es decir, debe existir una interoperabilidad que llegue hasta las extensiones privadas de la MIB.

Por defecto, las estaciones de gestión sólo conocen la MIB estándar. Por lo tanto, para poder gestionar objetos MIB privados es necesario que previamente se cargue la estructura de la MIB privada en la plataforma de gestión.

Las plataformas de gestión actuales permiten compilar nuevas MIB para gestionar recursos privados.

2.4. El protocolo SNMP

Existen unos conceptos básicos relacionados con las operaciones del protocolo SNMP:

- Las operaciones que soporta SNMP
- Las comunidades
- La identificación de instancias

2.4.1. Operaciones en SNMP

Todas las operaciones que soporta el protocolo SNMP están relacionadas con la modificación e inspección de variables. En los objetos escalares se pueden agrupar en tres tipos de operaciones básicas:

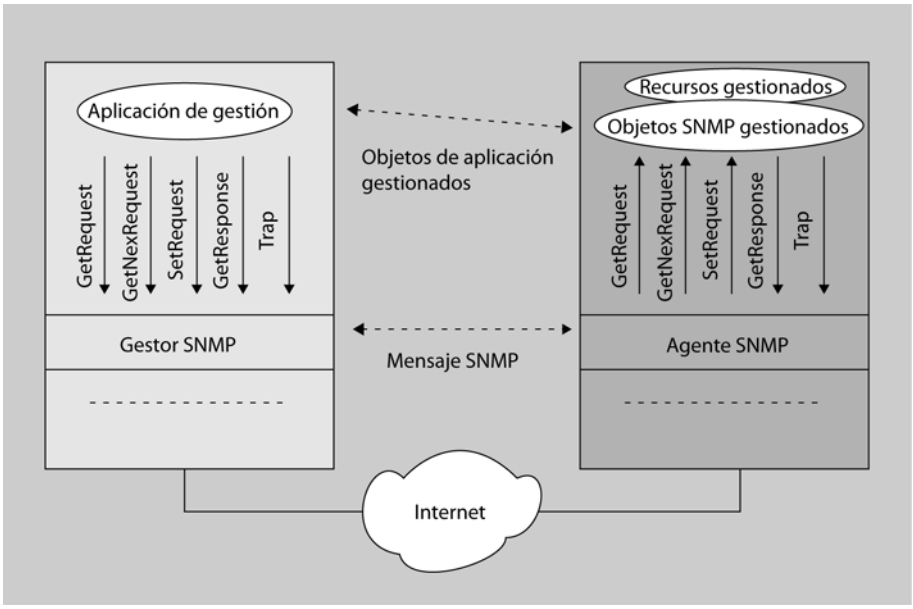
Get. La estación de gestión pide un objeto escalar a la estación gestionada.

Set. La estación de gestión actualiza un objeto escalar de una estación gestionada.

Trap. La estación gestionada envía el valor de un objeto escalar al gestor sin solicitud previa.

Se dan cinco tipos de mensajes usados por SNMP, como muestra la figura 8.

Figura 8. Acciones SNMP



Acceso a un único objeto cada vez

El acceso a un único objeto cada vez es una restricción que simplifica la implementación de SNMP, pero limita la capacidad del sistema de gestión.

No es posible acceder a una tabla entera con una única acción.

2.4.2. Las comunidades SNMP

La comunidad SNMP es una relación entre el agente y el gestor que define básicamente la **autenticación** y el **control de acceso**.

Una comunidad SNMP es una **relación** entre el **agente SNMP** y los **gestores SNMP**. El concepto de comunidad es **local** y se define en el sistema gestionado. Éste crea una comunidad para cada combinación de autenticación y control de acceso. A cada comunidad se le da un nombre de comunidad único (dentro del agente) que se proporciona en la estación de gestión. Esta estación,

por su parte, debe utilizar este nombre cada vez que realice una operación *get* o *set*. El agente puede establecer varias comunidades en función de los permisos de los gestores.

Entonces, **¿cómo lleva a cabo la autenticación?** Cada mensaje que genera el gestor hacia el agente ha de incluir el nombre de comunidad (*community name*).

Autenticación y control de acceso

Entendiendo por autenticación el caso en el que el mensaje es auténtico.

Control de acceso = diferentes privilegios en función del gestor.

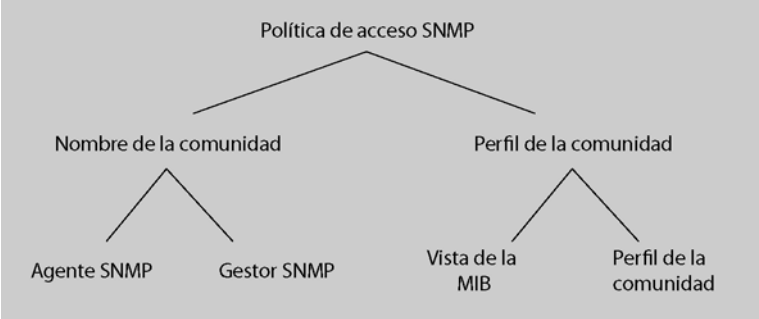
Y **¿cómo funciona la política de acceso?** El agente puede crear diferentes categorías de acceso a la MIB en función de quién sea el gestor. Lo que hace es crear diferentes perfiles ligados al nombre de comunidad que tienen acceso a las diferentes partes de la MIB con permisos concretos de lectura o lectura/escritura (modo de acceso).

Así, la figura 9 intenta mostrar todo lo que serían las políticas de acceso a SNMP. Por una parte, tenemos el nombre de comunidad que conocen tanto el agente, como el gestor. Por otra, lo que tenemos es que cada comunidad tiene asociado un perfil que le permite acceder a unas partes de la MIB concretas.

Nombres de comunidad de los dispositivos

Los dispositivos por defecto suelen llevar consigo dos nombres de comunidad conocidos por todos: *public* (lectura) y *private* (lectura/escritura). Es preciso tener en cuenta que se deben eliminar.

Figura 9. Conceptos administrativos



2.4.3. Identificación de instancias

Cada objeto en la MIB posee un único identificador de objeto, que se halla definido por la posición que ocupa dentro del árbol MIB. No obstante, cuando se realiza un acceso a la MIB vía SNMP lo que se quiere no es el tipo del objeto, sino una instancia concreta (el valor de la variable). Veamos cómo se lleva a cabo.

Cabe recordar la diferencia entre objeto e instancia del objeto.

Caso de objetos escalares

Para obtener la instancia de un objeto escalar es necesario marcar el camino para llegar al objeto e incluir siempre al final **.0**.

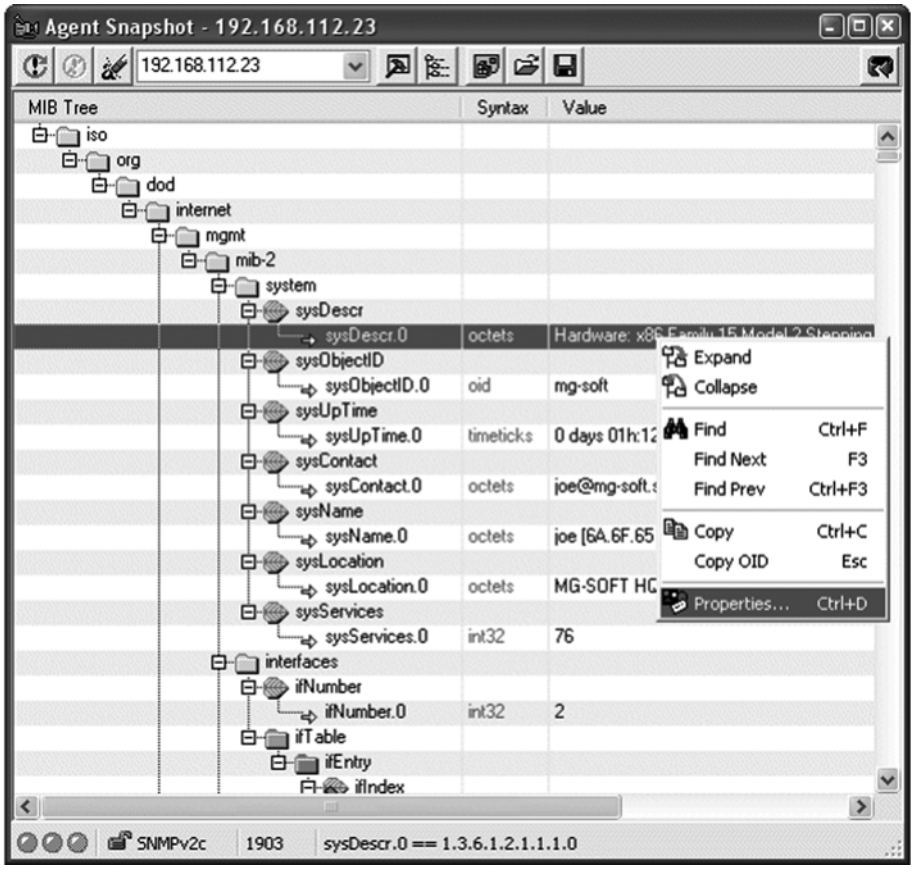
El 0 se pone para diferenciar entre realizar una referencia a un objeto y hacerla a la variable asociada a este objeto.

Por ejemplo tenemos,

	OBJECT IDENTIFER
Snmp subtree	1.3.6.1.2.1.11
snmpInBadValues	1.3.6.1.2.1.11.10
instancia de snmpInBadValues	1.3.6.1.2.1.11.10.0

En la figura 10 se muestra el árbol MIB-II y unas peticiones en los diferentes objetos del grupo *system*.

Figura 10. Resultado obtenido del grupo *system* con una aplicación de gestión



Objetos de tipo tabla

El acceso a las instancias dentro de una tabla, aunque el mecanismo es sencillo, resulta un poco complicado de explicar. Lo mejor es verlo mediante un ejemplo.

Ejemplo

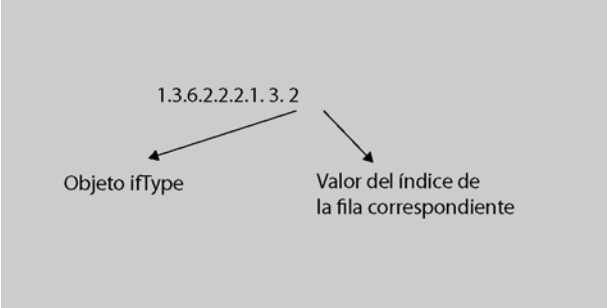
Supongamos un dispositivo con dos interfaces y que queremos información del grupo dentro de la MIB-II denominada *interface*. Este grupo tiene la siguiente forma:

Interfaces
- ifTable
  - ifEntry
    - ifDescr
    - IfIndex
    - ifType
    - ifMTU
    - ifSpeed

En este caso tendríamos una tabla con dos filas y cinco columnas de la forma siguiente:

ifDescr	ifIndex	ifType	ifMTU	ifSpeed	
					ifEntry (1.3.6.1.2.2.2.1)
					ifEntry (1.3.6.1.2.2.2.1)

A partir de esta información, si lo que queremos realizar es una instancia concreta, por ejemplo de *ifType*, deberemos efectuar:



En el caso de que tengamos una tabla con más de un índice se debería realizar:

Ruta.objeto.valor_indice1.valor_indice2_..valor_indice*n*

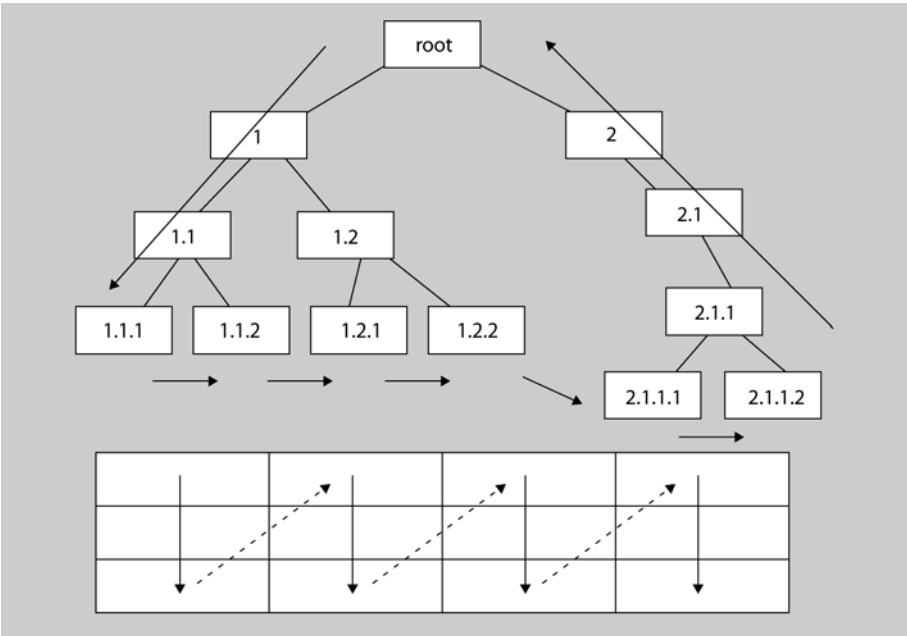
Ésta es la única forma de identificar correctamente la fila exacta donde se encuentra la instancia del objeto solicitado.

Conviene que tengáis presente que los índices pueden tener cualquier valor en función del objeto que representen.

Es muy interesante tener claro el orden lexicográfico en un árbol de objetos, tanto escalares como en una tabla.

En la figura 11 se muestra cómo nos movemos dentro del árbol jerárquico, y a la vez dentro de una tabla concreta, si vamos solicitando los valores de los objetos uno detrás de otro.

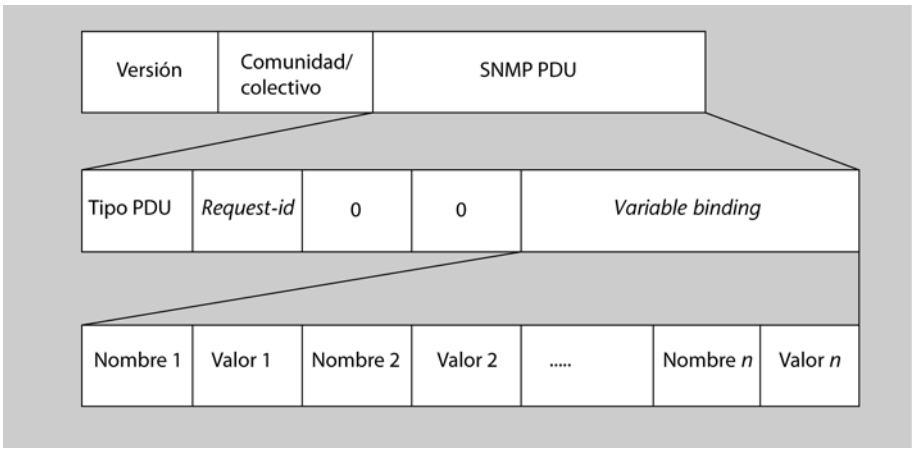
Figura 11. Orden de acceso a las instancias de los objetos dentro de las tablas SNMP



2.4.4. Formato del paquete SNMP

En la figura 12 se muestra el formato del paquete SNMP. Como podemos ver, todos los mensajes incluyen el número de versión de SNMP, nombre de la comunidad para utilizar en el intercambio y tipo de mensaje.

Figura 12. Formato de los paquetes SNM



Es conveniente saber que en un mismo mensaje SNMP se pueden incluir varias operaciones del mismo tipo.

Un caso interesante es *GetNextRequest*. Es similar a *GetRequest* pero con la diferencia de que permite ver el valor siguiente dentro de la estructura del árbol. Es decir, que cuando solicitamos una instancia no obtenemos el valor de la misma instancia, sino el valor correspondiente al objeto inmediatamente posterior al solicitado. Es muy útil en el caso de que se quiera acceder a los valores de una tabla. Si recordáis, hemos explicado que para acceder a las instancias de una tabla había que poner los valores de las instancias índice de esta tabla, lo cual es complicado *a priori*, ya que no conocemos estos valores. En este caso, lo que hacemos es pedir el valor justamente anterior dentro del árbol de objetos, y lo que el agente nos devuelve es el primer valor de la tabla.

Ejemplo

Si recordamos el ejemplo anterior a la petición siguiente: GetNext (1.3.6.1.2.2.1.0), la respuesta que obtendremos será:

GetReponse = 1.3.6.1.2.2.2.eth0.1

En este caso nos retorna el valor de *IfDescr* y a la vez el valor que tiene el índice de la primera fila.

2.5. Limitaciones del protocolo SNMP

Una vez visto el funcionamiento básico del protocolo SNMP, es necesario tener presente algunas de sus limitaciones:

- SNMP no es adecuado para la gestión de redes muy grandes, teniendo en cuenta las limitaciones de rendimiento ligadas al paradigma petición-respuesta.

- SNMP no está bien diseñado para grandes volúmenes de datos como son las tablas.
- Los *traps* no poseen reconocimiento, lo cual puede provocar que *traps* críticos se puedan perder sin que lleguen al gestor. Recordad que los *traps* permiten al agente generar un mensaje hacia el gestor sin petición previa.

¿Por qué los *traps* no tienen reconocimiento?

Porque, como se ha dicho, SNMP tiene UDP como protocolo de capa de transporte.

- La autenticación, como se ha visto, es muy trivial.
- La MIB tiene limitaciones.
- SNMP no soporta comunicación entre gestores.

3. SNMPv2

El SNMPv 2 es una evolución de la versión inicial de SNMP, la cual incluye toda una serie de mejoras con respecto a la versión anterior.

3.1. Mejoras en la gestión

Es necesario que tengáis presente que lo primero que hay que destacar es el cambio en la filosofía de trabajo de esta nueva versión. SNMPv2 puede soportar tanto **gestión centralizada** como **distribuida**. En este caso, algunos gestores realizarán el papel de gestor y agente simultáneamente. Como agentes aceptarán pedidos de un gestor de categoría superior. Estas peticiones pueden ser para obtener información del gestor intermedio o para pedir a este gestor un informe de los agentes que están subordinados a él. Asimismo, el gestor intermedio puede enviar *traps* al superior.

Las mejoras de SNMPv 2 se agrupan en tres categorías:

- Estructura de la información de gestión
- Capacidad gestor-gestor
- Operaciones del protocolo

3.2. Estructura de la información de gestión

La estructura de la información de gestión (SMI) crece con respecto a la SMI de la versión 1. Encontramos una estructura con más tipos de datos, mejora en la documentación asociada a cada objeto y un aspecto muy importante que es una nueva convención para crear y borrar filas en las tablas de la MIB.

Si hemos de resumir brevemente cómo queda esta estructura, se puede reducir a cuatro conceptos clave:

1) Definición de objetos. Se crea una nueva macro que permite incluir nuevos tipos de datos y mejorar la documentación asociada a cada objeto.

2) Tablas conceptuales. SNMPv2 mejora el mecanismo utilizado tanto en RFC 1212 como en la especificación RMON para facilitar la creación, eliminación y acceso a las filas de las tablas.

3) Definiciones de notificación. Se crea una macro específica que define la información enviada por una entidad SNMPv2 cuando se da un acontecimiento excepcional en la entidad.

4) Módulos de información. SNMPv2 introduce el concepto de módulo de información, que especifica un grupo de definiciones que se encuentran relacionadas.

3.3. El protocolo SNMPv2

En cuanto a las operaciones del protocolo, éste incluye dos nuevos tipos de PDU:

GetBulkRequest. Permite obtener un gran número de bloques de datos de forma eficiente, como son la posibilidad de obtener varias filas de una tabla.

InformRequest. Con esta PDU el gestor intermedio puede enviar *traps* a otro gestor.

4. RMON (*remote network monitoring*)

El objetivo de RMON es proporcionar información a la plataforma de gestión de los diferentes segmentos de red, pero sin cambiar el protocolo SNMP. Para realizarlo, podemos considerar que dentro de la red tendremos una serie de dispositivos que serán capaces de monitorizar la información, analizarla y comunicarnos de manera adecuada con el gestor. Ello, de entrada, evita la necesidad de conocer el tráfico de entrada y salida de cada uno de los dispositivos, lo cual, en la mayoría de casos, no es interesante.

Tradicionalmente, los equipos que han analizado el tráfico han sido los analizadores de red o también llamados en ocasiones denominados detectores (*sniffers*).

Actualmente, muchos de los conmutadores (*switchs*) de gama media y alta incorporan prestaciones RMON.

4.1. Objetivos del diseño de RMON

Los objetivos del diseño de RMON se hallan especificados en el RFC 1757 y son básicamente los siguientes:

Limitar el sondeo (*polling*) permitiendo llevar a cabo operaciones fuera de línea. Conseguimos mejorar el rendimiento, continuar recopilando información del segmento en caso de que caiga la línea entre gestor y monitor, y, cuando ésta se recupera, enviar la información.

Monitorización proactiva. Si el monitor posee bastantes recursos, es interesante que pueda sacar diagnosis e informes de rendimiento de la red.

Detección de problemas y envío de informes (*reports*).

Posibilidad de diálogo con más de un mánager. Esto mejora la fiabilidad.

Análisis del tráfico recogido.

4.2. Control de la sonda RMON

El dispositivo que soporta RMON se denomina normalmente sonda RMON o simplemente **sonda**.

La sonda RMON posee alguna característica propia a la hora de ser gestionada, ya que es capaz de efectuar trabajos más complejos que un agente con MIB-II. Estas características estarían asociadas a la **configuración de la sonda** y a la **invocación de acciones**.

4.2.1. Configuración de la sonda

Una sonda necesita ser configurada para recopilar datos. En función de la configuración que realicemos nos permitirá decidir el tipo y la forma de los datos recogidos. La manera de llevarlo a cabo se explica a continuación.

La MIB RMON está estructurada en grupos como MIB-II, pero con la particularidad de que todos los grupos están formados por tablas. Así, para cada grupo existen **tablas de control** y **tablas de datos**. Las tablas de control son tablas de lectura/escritura y contienen parámetros que describen los datos. Las tablas de datos son sólo de lectura. Si lo pensáis, tiene sentido que estas tablas sólo sean de lectura. Lo que lleva a cabo la estación de gestión es configurar los parámetros de control apropiados para que la sonda recoja los datos que interesan al gestor en cada momento.

Parece lógico que la MIB tenga sólo tablas, ya que lo que hace es obtener información de los segmentos de la red.

Los parámetros se ponen incluyendo una nueva fila o modificando una fila de la tabla de control. Por lo tanto, las funciones que debe efectuar una sonda se encuentran definidas e implementadas en términos de filas de una tabla.

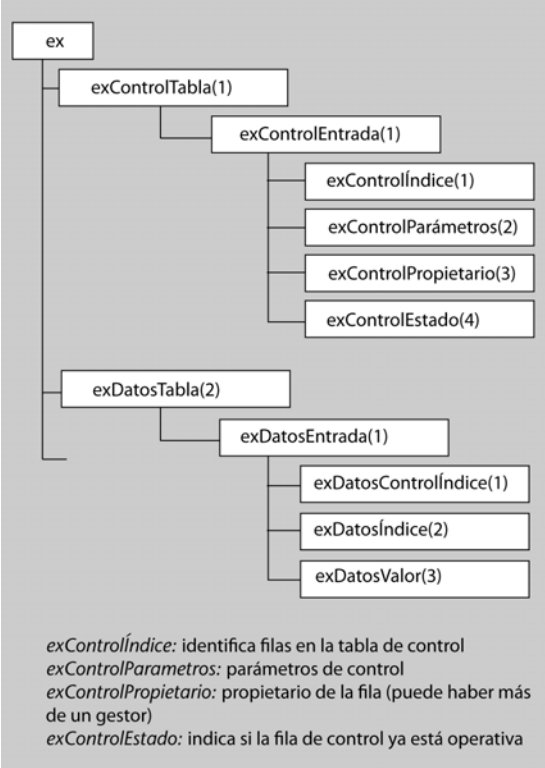
Por ejemplo, en la tabla de control se pueden indicar parámetros como el origen de los datos, el tipo, el tiempo entre capturas, etc. Así, el índice en la tabla de control sirve para acceder a la tabla de datos, y al revés.

La mejor manera de entender el funcionamiento es mediante un ejemplo.

Ejemplo de configuración de una sonda

Supongamos el grupo ficticio *ex* (ejemplo) con la estructura que se muestra en la figura 13.

Figura 13. Representación de un grupo genérico siguiendo el formato RMON



La tabla de datos tiene dos índices, donde *exDadesControlIndex* coincide con *exControlIndex*.

Así, una posible configuración de las tablas se muestra en la figura 14.

Figura 14. Valores concretos de una posible tabla resultante

exControlÍndice	exControlParámetro	exControlPropietario	exControlEstado
1	5	gestor1	válido(1)
2	26	gestor2	válido(1)
3	19	gestor2	válido(1)

exDatosControlÍndice	exDatosÍndice	exDatosValor
1	1	25
2	1	34
2	2	45
2	3	12
3	1	11
3	2	19
3	3	23

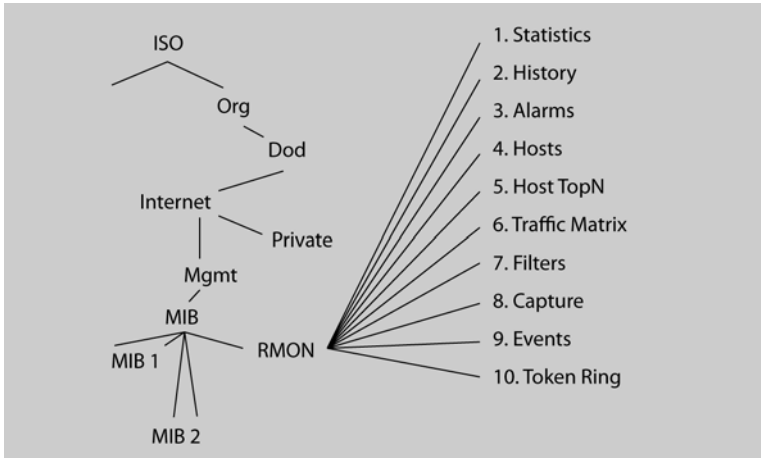
Cada fila de la tabla de control llama a unas tablas para que guarden los valores.

El índice común de las dos tablas permite identificar las filas de datos asociadas a un gestor concreto creado en la tabla de control.

4.3. RMON1 y RMON2

Inicialmente, RMON fue desarrollado para redes LAN Ethernet y Token Ring, de manera que divide las funciones de monitorización en nueve grupos asociados a topologías Ethernet y un grupo diez para Token Ring. En la figura 15, se ven los diez grupos que forman parte de la rama MIB.

Figura 15. Grupos RMON1 dentro de la estructura MIB



El estándar RMON está aceptado extensamente por las empresas de *networking* y comunicaciones.

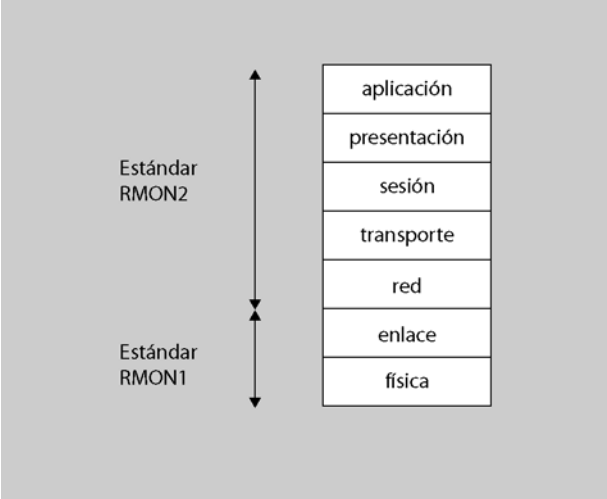
Una limitación importante de RMON1 es que especifica sólo monitorización y diagnóstico del tráfico de red en el nivel de capa de enlace y no realiza la monitorización del tráfico en el nivel de aplicaciones de extremo a extremo.

Como consecuencia de esta limitación nos encontramos con que RMON1 no puede identificar dispositivos que se hallen fuera de su subred, es decir, detrás del encaminador.

Pensad atentamente en lo que ello implica.

RMON2 responde a la necesidad de analizar el tráfico de las capas superiores. Se puede considerar una extensión de RMON1, y son tecnologías complementarias, como se puede ver en la figura 16.

Figura 16. Capas OSI donde aplica cada uno de los protocolos RMON1 y RMON2



En la tabla siguiente podemos ver cómo se complementan ambos estándares para cubrir los diferentes temas de monitorización de red.

Aspectos de la gestión	Nivel OSI	Estándar RMON
Utilización y errores físicos	Control de acceso al medio	RMON1
Segmentación LAN	Enlace	RMON1
Interconexión de redes	Red	RMON2
Uso de aplicación	Aplicación	RMON2

4.3.1. La MIB RMON

Como se ha comentado, la MIB RMON se halla dividida en diez grupos. En la tabla siguiente se muestra una descripción de cada uno de los grupos.

Grupo	Descripción
Statistics	Mantiene estadísticas de utilización y errores de los niveles bajos para cada subred monitorizada.
History	Guarda muestras de estadísticas de manera periódica. Las extrae del grupo *statistics.*
Alarm	Permite al gestor configurar intervalos de muestra y umbrales de alarma asociados a los valores obtenidos por la sonda.
Host	Contiene contadores de varios tipos de tráfico de los *hosts* de la subred.
hostTopN	Contiene estadísticas de los *hosts* ordenadas.
Matrix	Muestra información de errores y utilización en forma de matriz.
Filter	Permite ver ciertos paquetes a partir de un filtro concreto.
Capture	Controla cómo se envían los datos a la consola de gestión.
Event	Ofrece una tabla con los acontecimientos generados en la sonda.
TokenRing	Mantiene estadísticas e información de configuración para las subredes Token Ring.

A su vez, los grupos RMON los podemos agrupar en función de la tarea que desempeñan. Podemos encontrar dos grupos de tareas básicas:

- Estadísticas de tráfico
- Alarmas, filtros y captura de paquetes

Alarmas, filtros y captura de paquetes

Aunque dentro del grupo de alarmas, filtros y captura de paquetes existen tres funciones, veremos que dependen unas de las otras para su correcto funcionamiento.

Todos los grupos son opcionales en su implementación, sin embargo se dan dependencias entre algunos de ellos. Así, las dependencias principales son las siguientes:

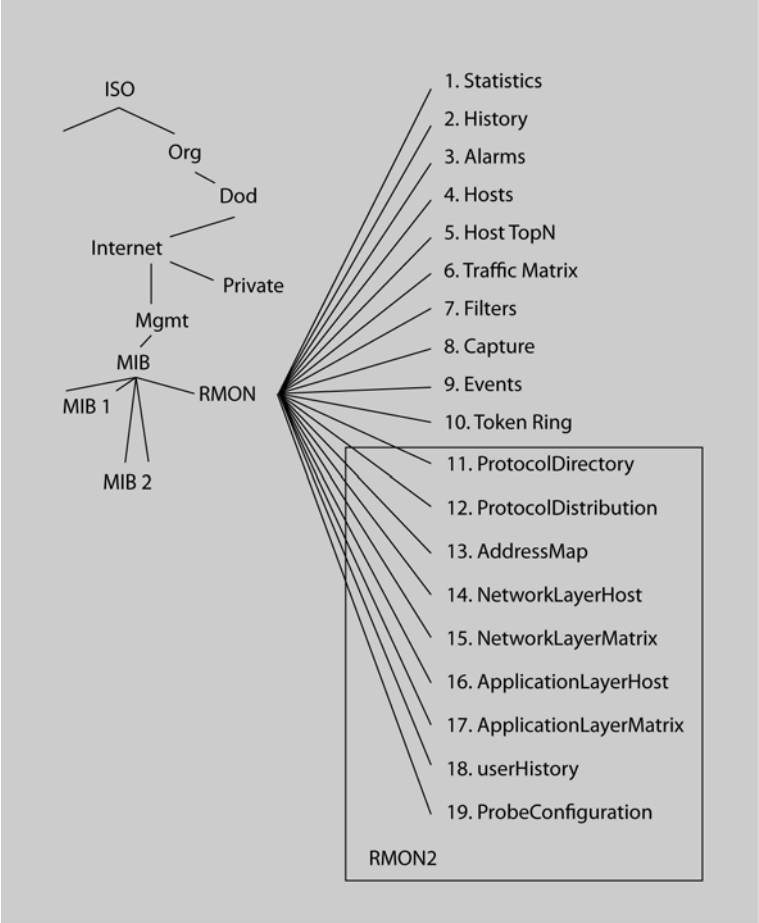
- El grupo *alarm* requiere la implementación del grupo *event*.
- El grupo *hostTopN* requiere la implementación del grupo *host*.
- El grupo *capture* requiere la implementación del grupo *filter*.

4.3.2. RMON2

Como hemos comentado anteriormente, **RMON2** descodifica paquetes de **nivel 3 a 7** de la pila OSI. Ello significa que puede realizar **monitoreo de tráfico en el nivel IP** y, por lo tanto, no está limitado a un segmento Ethernet como pasaba en RMON1. Podemos saber hacia dónde se dirige el tráfico que va hacia fuera de la propia subred, y lo mismo sucede con el tráfico entrante. Hasta ahora este tráfico en el nivel 2 siempre tenía la dirección MAC del encaminador por defecto. Lo mismo ocurrirá con la descodificación en el nivel de aplicación. Podremos saber hacia qué aplicación se genera el tráfico de cada una de las subredes.

Tal y como se puede ver en la figura 17, la MIB RMON2 es una extensión de la MIB RMON.

Figura 17. Grupos RMON2 dentro de la estructura MIB



Otras características de RMON2

Las nuevas características de RMON 2 se encuentran asociadas al modo de **indexar de las tablas**. Un gestor lo que lleva a cabo es solicitar la información de las sondas de manera periódica con el inconveniente, en cuanto a eficiencia, que implica el hecho de que cada vez devuelve los nuevos valores además de los antiguos. Lo que incorpora RMON2 es un índice temporal que permite enviar sólo los valores de los objetos que han cambiado desde la última petición en la sonda RMON.

RMON2 no envía tantos datos y, por tanto, lleva a cabo un uso más eficiente del ancho de banda.

5. Gestión TMN

Las operadoras de telecomunicaciones requieren una infraestructura de gestión singular. Este entorno de gestión ha de prever como objetivo último la prestación de servicio de manera flexible y dinámica. El modelo clásico de gestión orienta sus tareas hacia la red, es decir, hacia la configuración, el mantenimiento y, si es posible, el análisis de rendimiento. La necesidad de las capacidades de gestión respecto a los clientes y los servicios ha provocado la aparición de modelos de referencia para desarrollar la gestión TMN.

Hace unos años la industria de las telecomunicaciones utilizaba soluciones propietarias para la gestión. Las operadoras daban nuevos servicios en función de sus posibilidades y, por lo tanto, iban mejorando su sistema de gestión. Actualmente, con la desregularización del sector, el incremento de la competencia y los nuevos servicios las operadoras se han encontrado con que la gestión propietaria no proporciona interoperabilidad entre las diferentes tecnologías y las soluciones de gestión que incorporan. Otro aspecto que deben tener en cuenta es la coexistencia de las nuevas soluciones con sistemas "antiguos".

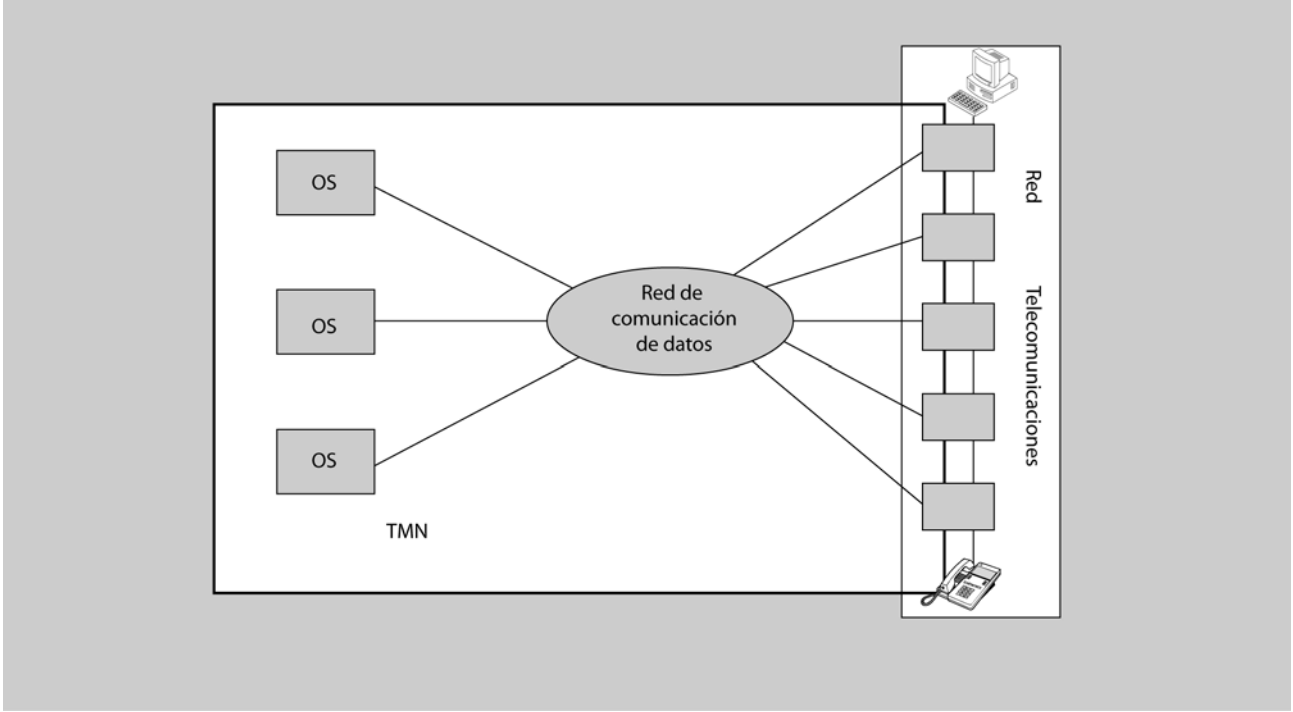
Por lo tanto, y a modo de resumen, podemos decir que la motivación de la arquitectura TMN ha sido, por una parte, la heterogeneidad de las redes de telecomunicaciones y, por otra, las demandas sobre aspectos como los siguientes:

- Posibilidad de introducir nuevos servicios
- Alta calidad de los servicios
- Posibilidad de reorganizar las redes
- Métodos eficientes de trabajo para operar las redes
- Competencia entre operadoras

Los principios del TMN (*telecomunications management networks*) se tratan en la recomendación M.3010 de la ITU-T, que **define un modelo de operación por capas**: capas de operación. Tiene una fuerte relación con el modelo de gestión OSI.

El **objetivo de TMN** es proporcionar una estructura de red organizada para conseguir la interconexión de los diferentes tipos de sistemas de operación y equipos de telecomunicación usando una arquitectura estándar e interfaces normalizadas.

Figura 18. Relación de TMN en una red de telecomunicaciones



A diferencia del modelo OSI que definía cinco áreas funcionales, el estándar TMN no entra en consideraciones sobre aplicaciones de la información gestionada. Por el contrario, se definen las funciones siguientes:

Podéis consultar las cinco áreas funcionales del modelo OSI en el subapartado 1.1.1 de este módulo.

- El intercambio de información entre la red gestionada y la red TMN
- El intercambio de información entre redes TMN
- La conversión de formatos de información para un intercambio consistente de la información
- La transferencia de información entre puntos de una TMN
- El análisis de la información de gestión y la capacidad de actuar en función de ésta
- La manipulación y presentación de la información de gestión en un formato útil para el usuario de la misma información
- El control de acceso a la información de gestión por parte de los usuarios autorizados

No olvidéis que el significado de gestión de red para TMN es más amplio que el que habíamos considerado en el caso de SNMP. En ambos casos, gestión de red significa gestión de redes y servicios, pero en TMN se halla focalizado en redes de telecomunicaciones, equipos y servicios proporcionados a los clientes.

Protocolos que cubre TMN

Algunos ejemplos de protocolos o tecnologías que cubre TMN serían: ATM, SDH, SONET o xDSL.

En la industria de las telecomunicaciones, los servicios y las configuraciones de los usuarios finales (clientes) se encuentran incluidos como parte de TMN. Así, aparecen dos términos nuevos, **servicio** y **aprovisionamiento de recursos**.

TMN proporciona el soporte de gestión para la planificación, aprovisionamiento, instalación, mantenimiento, operación y administración de redes y servicios de telecomunicaciones.

Una red de telecomunicaciones puede incluir una gran variedad de componentes, como equipos para transmisión analógica, digital o sin hilos (*wireless*), etc.

5.1. Arquitectura TMN

La arquitectura TMN está dividida en tres bloques. La recomendación M.3010 define los siguientes bloques:

Arquitectura física. Estructura y entidades de la red: cómo se implantan las funciones de gestión en los equipos físicos.

Arquitectura funcional. Componentes y funciones de gestión.

Arquitectura de la información. Niveles de gestión: la gestión se estructura según responsabilidades.

Dirección recomendada

Podéis encontrar un interesante *tutorial* sobre la arquitectura TMN en http://www.simpleweb.org/tutorials/tmn/

5.1.1. Arquitectura funcional del TMN

La arquitectura funcional TMN divide el dominio TMN en diferentes bloques funcionales. Cada bloque funcional ejecuta una función específica de gestión. Los bloques son los siguientes:

Operations system function **(OSF)**. Proporciona las funciones de planificación y gestión para la red de telecomunicaciones y para los mismos componentes.

Network element function **(NEF)**. Monitorizado y controlado por TMN.

Workstation function **(WSF)**. Permite que el usuario pueda ver la información.

Mediation function **(MF)**. Una especie de pasarela para el intercambio de información de gestión cuando los bloques funcionales poseen diferentes puntos de referencia.

Q adaptor function **(QAF)**. Para trasladar información de gestión entre puntos de referencia TMN y no TMN.

Un elemento importante que se ha mencionado pero que todavía no se ha comentado es el de los puntos de referencia (*reference points*). Estos puntos son los límites conceptuales de los diferentes bloques.

5.1.2. Arquitectura física del TMN

La arquitectura física explica la implementación de los bloques funcionales en sistemas físicos y la interfaz entre ellos. Los componentes de la arquitectura física son:

Operations system **(OS)**. Es el equivalente al gestor. Ofrece soporte para el procesamiento de información relacionada con operaciones, administración, mantenimiento y aprovisionamiento de las redes de telecomunicaciones.

Data communications network **(DCN)**. Capacidad de encaminamiento y transporte para el intercambio de información entre OS y OS, OS y NE, WS y OS, y WS y NE.

Mediation device **(MD)**. Tiene funciones de retransmisión o pasarela.

Workstation **(WS)**. Punto de entrada o salida que permite a los operadores del sistema acceder a los datos de gestión.

Network element **(NE)**. Lo que conocemos por agente.

Q adaptor **(QA)**. Convierte los datos no TMN en datos TMN, y viceversa.

5.1.3. Arquitectura de información del TMN

La arquitectura de información explica cómo los sistemas de gestión OSI y los principios X.500 pueden ser aplicados a TMN. La arquitectura de información describe los recursos que deben ser gestionados por TMN utilizando las guías de referencia para la definición de los objetos gestionados y la sintaxis ASN.1

Resumen

En este módulo didáctico se ha intentado dar una breve visión de conceptos básicos de la gestión de red y el porqué de su necesidad en un entorno como el actual. Para ello, se han explicado los conceptos generales de la gestión y, posteriormente, se ha profundizado en los protocolos más usados.

Se ha procurado separar lo que es la gestión de red, en lo que hemos denominado redes Internet, de lo que son las redes de las operadoras o redes de telecomunicaciones.

Como se ha visto, el protocolo dominante en el mundo Internet es el protocolo SNMP, que ha ido evolucionando para cubrir las diferentes necesidades que han ido surgiendo.

Inicialmente, SNMPv1 era un protocolo sencillo que trabajaba de forma centralizada con el paradigma petición-respuesta. Después, de la misma manera que las redes se hacen más complejas, el protocolo ha tenido que evolucionar para solucionar otros aspectos no planteados al principio. Así, SNMPv2 incluye mecanismos que permiten optimizar la información almacenada en las tablas de los distintos agentes. Finalmente, aunque no se ha comentado, hay una nueva versión SNMPv3 que lo que introduce son mecanismos de seguridad.

Al mismo tiempo, dentro de la MIB se ha incorporado RMON, que como ya habéis visto permite monitorizar las redes sin tener que recurrir a solicitar información de cada uno de los dispositivos vía SNMP.

Por último, para cerrar el módulo, se han descrito las necesidades de las operadoras de telecomunicaciones y las posibilidades que les ofrece el estándar TMN para cubrir sus necesidades de gestión que, en la mayoría de casos, son bastante diferentes de las necesidades de las redes convencionales.

Actividades

1. Descargad de Internet el RFC 1212 y el RFC 1213, y familiarizaros con la notación ASN.1 utilizada para definir las diferentes MIB.

2. Realizad un *ping* en un dispositivo de la red donde os encontréis e intentad determinar los valores obtenidos.

3. Con la ayuda de MG SOFT MIB Browser intentad determinar los diferentes objetos de la MIB-II. Podéis encontrar una versión de evaluación de este programa en http://www.mg-soft.com

4. Pensad el motivo por el que existen tablas SNMP que necesitan más de un índice.

5. Averiguad cuáles son las plataformas de gestión de dos fabricantes como pueden ser HP y Cisco.

6. Razonad por qué motivo se optó por trabajar con UDP y no TCP en gestión SNMP.

Ejercicios de autoevaluación

1. Tenemos las peticiones del grupo *rs232* siguientes:

GetNext rs232.rs232Number.0
Response: rs232.rs232InSigTable.rs232InSigEntry.rs232InSigPortIndex.1.1 = 1

Getnext rs232.rs232InSigTable.rs232InSigEntry.rs232InSigPortIndex.1.1
Response rs232.rs232InSigTable.rs232InSigEntry.rs232InSigPortIndex.2.2 = 2

Getnext rs232.rs232InSigTable.rs232InSigEntry.rs232InSigPortIndex.2.2
Response: rs232.rs232InSigTable.rs232InSigEntry.rs232InSigState.1.1 = 2

Getnext rs232.rs232InSigTable.rs232InSigEntry.rs232InSigState.1.1
Response rs232.rs232InSigTable.rs232InSigEntry.rs232InSigState.2.2 = 2

Getnext rs232.rs232InSigTable.rs232InSigEntry.rs232InSigState.2.2
Reponse rs232.rs232InSigTable.rs232InSigEntry.rs232InSigChanges.1.1 = 3

Getnext rs232.rs232InSigTable.rs232InSigEntry.rs232InSigChanges.1.1
Response Reponse rs232.rs232InSigTable.rs232InSigEntry.rs232InSigChanges.2.2 = 4

A partir de la información dada por estas peticiones os pedimos lo siguiente:
a) Forma de la tabla (contenido de la tabla).
b) ¿Cuáles son los objetos índice?
c) ¿Si hacemos un *GetNext* a partir del último valor obtenido, qué objeto obtendremos?
d) Realizad el árbol del objeto definido.

Nota: El nombre del segundo índice es rs232InSigIndex.

2. Buscad el RFC 1213 donde está definida la MIB-II y a partir de ésta determinad el árbol del grupo UDP.

Solucionario

1.

a) En este ejercicio observamos que se ejecuta un *Getnext*. Se ejecuta porque queremos acceder a una tabla SNMP. Como se ha explicado, para hacer la petición hay que saber el valor de los índices correspondientes. Como eso no es así, con el *Getnext* la respuesta nos da el primer valor de la tabla y de manera indirecta los valores (contenido) de los índices de la primera fila de la tabla.

La primera respuesta

Response: rs232.rs232InSigTable.rs232InSigEntry.rs232InSigPortIndex.1.1 = 1

nos dice:

Primer valor de la tabla *rs232InSifPortIndex = 1* y además nos dice que la tabla tiene dos índices con valores 1 y 1.

A partir de aquí, los sucesivos *Getnext* van recorriendo toda la tabla de la misma manera que se muestra en la figura 11. El resultado es el siguiente:

InSIgPortIndex	InSigState	InSigChanges	InSigIndex
1	2	3	1
2	2	4	2

b) Los objetos índice son *InSIgPortIndex* y *InSigIndex*.

c) Se obtendría el valor *de InSigIndex* de la primera fila de la tabla.

d)
```
rs232
   rs232InSigTable
      rs232InSigEntry
         rs232InSigPortIndex (1)
         rs232InSigState (2)
         rs232InSigChanges (3)
         rs232InSigIndex (4)
```

2. El grupo UDP en el RFC 1213 tiene la siguiente forma:

```
udpInDatagrams OBJECT-TYPE
 SYNTAX Counter
 ACCESS read-only
 STATUS mandatory
 DESCRIPTION
 "The total number of UDP datagrams delivered to
 UDP users."
 ::= { udp 1 }

udpNoPorts OBJECT-TYPE
 SYNTAX Counter
 ACCESS read-only
 STATUS mandatory
 DESCRIPTION
 "The total number of received UDP datagrams for
 which there was no application at the destination
 port."
 ::= { udp 2 }

udpInErrors OBJECT-TYPE
 SYNTAX Counter
 ACCESS read-only
 STATUS mandatory
 DESCRIPTION
 "The number of received UDP datagrams that could
 not be delivered for reasons other than the lack
 of an application at the destination port."
 ::= { udp 3 }
```

```
udpOutDatagrams OBJECT-TYPE
 SYNTAX Counter
 ACCESS read-only
 STATUS mandatory
 DESCRIPTION
 "The total number of UDP datagrams sent from this
 entity."
 ::= { udp 4 }

-- the UDP Listener table

-- The UDP listener table contains information about this
-- entity's UDP end-points on which a local application is
-- currently accepting datagrams.

udpTable OBJECT-TYPE
 SYNTAX SEQUENCE OF UdpEntry
 ACCESS not-accessible
 STATUS mandatory
 DESCRIPTION
 "A table containing UDP listener information."
 ::= { udp 5 }

udpEntry OBJECT-TYPE
 SYNTAX UdpEntry
 ACCESS not-accessible
 STATUS mandatory
 DESCRIPTION
 "Information about a particular current UDP
 listener."
 INDEX { udpLocalAddress, udpLocalPort }
 ::= { udpTable 1 }

UdpEntry ::=
 SEQUENCE {
 udpLocalAddress
 IpAddress,
 udpLocalPort
 INTEGER (0..65535)
 }

udpLocalAddress OBJECT-TYPE
 SYNTAX IpAddress
 ACCESS read-only
 STATUS mandatory
 DESCRIPTION
 "The local IP address for this UDP listener. In
 the case of a UDP listener which is willing to
 accept datagrams for any IP interface associated
 with the node, the value 0.0.0.0 is used."
 ::= { udpEntry 1 }

udpLocalPort OBJECT-TYPE
 SYNTAX INTEGER (0..65535)
 ACCESS read-only
 STATUS mandatory
 DESCRIPTION
 "The local port number for this UDP listener."
 ::= { udpEntry 2 }
```

Si repasamos cada uno de los objetos de este grupo, nos queda una estructura como la siguiente:

```
  udp
    udpInDatagrams (1)
    udpNoPorts (2)
    udpInErrors (3)
    udpOutDatagrams (4)
    udpTable (5)
        udpEntry (1)
            udpLocalAddress (1)
            udpLocalPort (2)
```

Glosario

abstract syntax notation one *m* Lenguaje formal usado para definir sintaxis. En el caso de SNMP, la notación ASN.1 se usa para definir los objetos y el formato del protocolo.
sigla **ASN.1**

agente *m* En SNMP, módulo software que ejecuta las funciones de gestión de red que le pide el gestor. Gestor y agente se comunican vía SNMP.

ASN.1 *m* Véase ***abstract syntax notation*** *one*.

base de datos de información *f* En el contexto SNMP, contenedor de la información de los objetos para gestionar de manera estructurada donde cada variable representa un recurso.
sigla **MIB**

comunidad *f* En el contexto SNMP, relación entre agente y un grupo de gestores que define características de seguridad. El concepto de comunidad es local y se define en el agente.

estación de gestión *f* Módulo software que ejecuta aplicaciones de gestión para el control y monitoreo de elementos de red.
sin. **gestor**

gestor *m*
sin. **estación de gestión**

instancia de un objeto *f* Instancia específica de un tipo de objeto que está vinculado a un valor específico.

MIB *f* Véase **base de datos de información.**

objeto *m* Variable que representa algún recurso u otros aspectos del dispositivo gestionado.

remote network monitoring *m* Especificación de monitorización para analizar y monitorizar protocolos de red.
sigla **RMON**

RMON *m* Véase ***remote network monitoring***.

simple network management protocol *m* Colección de especificaciones para la gestión de red que incluye el protocolo, la definición de la estructura de datos y otros conceptos asociados.
sigla **SNMP**

SNMP *m* Véase ***simple network management protocol***.

sonda *f* En el contexto RMON, dispositivo de monitoreo remoto.

telecomunication management network *m* Conceptualmente, red separada que interactúa con la red de telecomunicaciones mediante varios puntos. El concepto de TMN está definido por la recomendación M.3010.
sigla **TMN**

TMN *m* Véase ***telecomunication management network***.

trap *m* Mensaje enviado por el agente hacia la estación de gestión sin que lo haya solicitado.

Bibliografía

Mauro, D. R.; Schmidt, K. J. (2001). *Essential SNMP*. Sebastopol, EE. UU.: O'Really.

(Disponible EN http://www.unix.org.ua/orelly/networking_2ndEd/snmp)

Stalling, W. (1998). *SNMP, SNMPv2, SNMPv3 and RMON 1 and 2*. Reading, Massachussets: Addisson-Wesley.

Subramanian, M. (2000). *Network Management: Principles and Practice*. Reading, Massachussets: Addison-Wesley.

Udupa, D. (1999). *TMN Telecomunications Management Network*. McGraw-Hill.