Administración y Gestión de Redes
Lic. Sistemas de información

Laboratorio de Redes,
Recuperación de Información
y Estudios de la Web

Protocolo SNMP

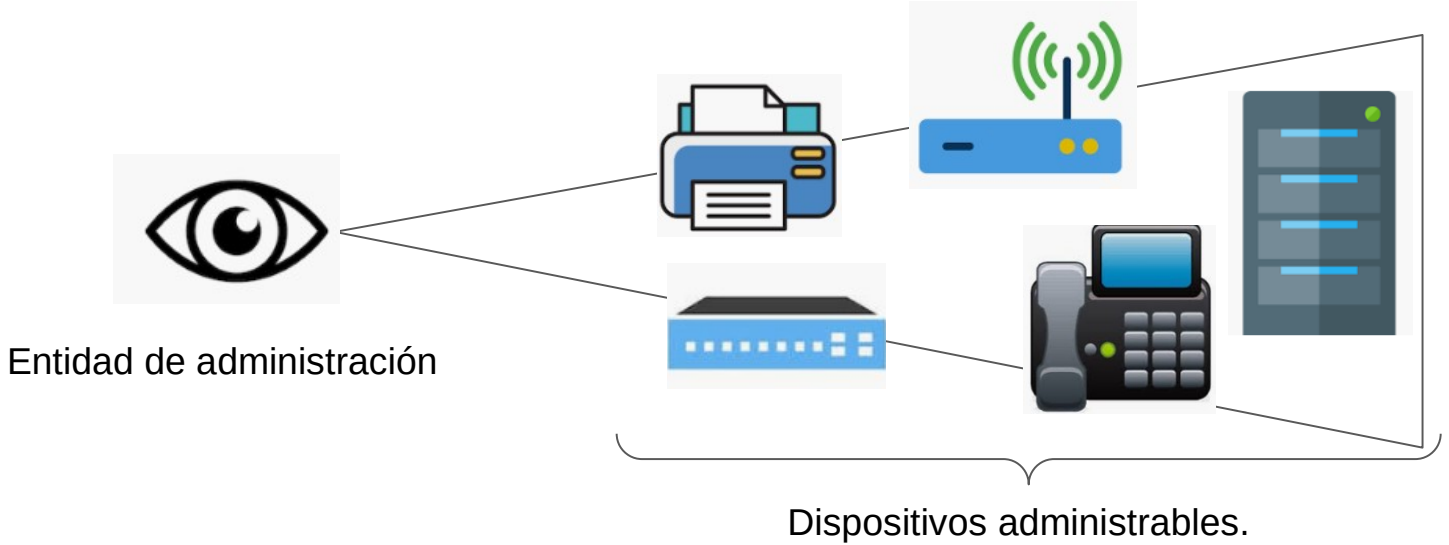
Simple Network Management Protocol

Equipo docente:
Fernando Lorge (florge@unlu.edu.ar)
Santiago Ricci (sricci@unlu.edu.ar)
Alejandro Iglesias (aaiglesias@unlu.edu.ar)
Mauro Meloni (maurom@unlu.edu.ar)
Patricio Torres (ptorres@unlu.edu.ar)

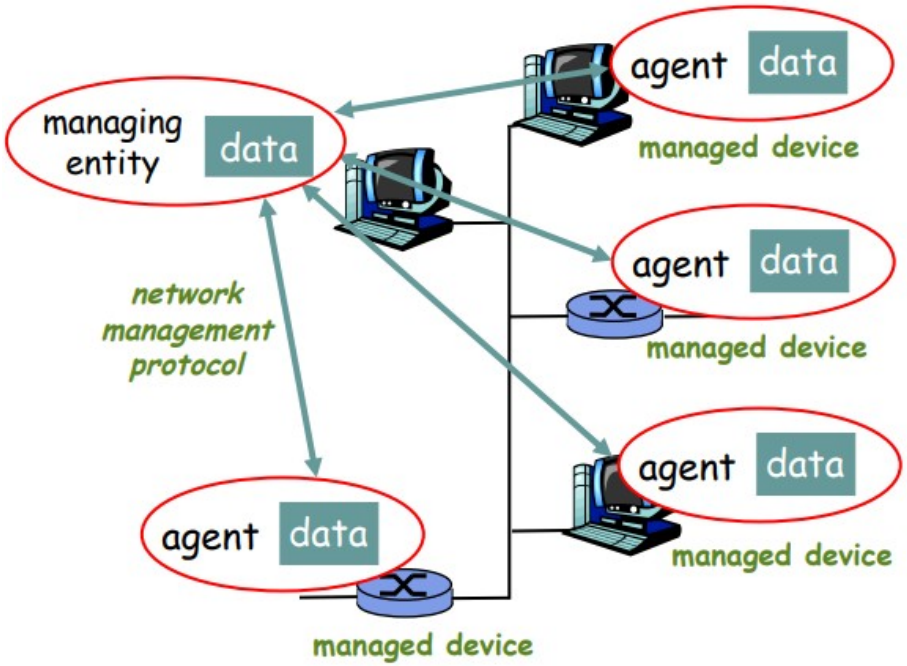
v2025

Herramientas de monitoreo y control

Existen diferentes herramientas y alternativas a la hora de monitorear una red. Distintas organizaciones y empresas han creado, además, protocolos que se pueden implementar en los dispositivos que componen una red para tal fin.



Arquitectura de un sistema de administración de red



Los **“Managed devices”**
(ej: switches, routers, ups, pcs..)

contienen

“Managed objects”
(ej: estado de la interfaz, tamaño de la tabla de rutas, temperatura en baterías)

cuyos datos son recolectados dentro de una

“Management Information Base”
(la información del estado de estos objetos organizada en un un árbol)

utilizando un
“Network Management Protocol”
(ej: snmp, netconf)

Estándares

Sigla	CMIP	SNMP	NETCONF
Organización	(OSI)	(IETF)	(IETF)
Nombre	Common Management Information Protocol	Simple Network Management Protocol	Network Configuration Protocol
Estándar	Recomendaciones ITU-T X.700, ISO/IEC 9596-1	Varias RFC's (desde fines de 80's)	RFC 6241 (2011)
Descripción	Diseñado en los '80 como el standard para gestión de redes	Inicialmente standard simple mientras se desarrollaban alternativas mejores	API formal expuesta por los dispositivos Utiliza el paradigma RPC Codificación XML
Adopción	Estandarización muy lenta	Desplegado y adoptado rápidamente	En rápida adopción
Características	Complejo y requiere mas recursos que snmp	Muy utilizado para monitoreo, configuración limitada	Seguridad y capacidad de configuración

¿Qué es SNMP?

Es un protocolo que facilita el intercambio de información de administración entre dispositivos de una red y una estación de monitoreo (alertas y estado del dispositivo).

- Es de capa de aplicación y utiliza **UDP** como protocolo de transporte.
- Simple y fácil de implementar, **posee diferentes versiones** (2c y 3 en uso)
- Permite **obtener información** de un dispositivo de forma remota y **modificar su configuración.**
- Organiza la información de un dispositivo en una **estructura de árbol** que denomina “**MIB**” (**M**anagement **I**nformation **B**ase).

Versiones del protocolo

Versión del protocolo	V1	v2 y v2c	V3
Fecha de creación	1988	1993 y 1996	2002
Objetivo	Lograr una solución temporal hasta la llegada de protocolos de gestión mejores y más completos	Reducir el overhead, lograr un mayor detalle en la definición de las variables, crear estructuras de datos	Mejorar la seguridad del protocolo
Mensajes	get, set	bulkwalk	framework de seguridad y mejoras en eficiencia
Se utiliza actualmente	no (en teoría)	si (muy extendido)	si

Cómo funciona el protocolo

Se organiza con arquitectura **cliente/servidor** y las **dispositivos administrables tienen un servidor en escucha** en el puerto por defecto 161.

- Los **administradores pueden enviar mensajes** a ellos y obtener una **respuesta en forma sincrónica** utilizando un cliente snmp.
- Los dispositivos organizan su información de estado utilizando una **MIB**, e identifica cada objeto (dato) mediante un número denominado **OID.**

Administrador

Cliente SNMP

¿Cómo te llamas?

get 1.3.6.1.2.1.1.5.0

"swtA"

Entidad administrable (switch)

Agente SNMP puerto 161

MIB

OID	Descripción	Contenido
1.3.6.1.2.1.1.5.0	sysName	swtA
1.3.6.1.2.1.2.1	ifNumber	5

¿Qué mensajes implementa el protocolo?

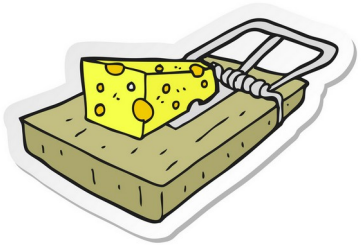
Obtener información	**GET** permite solicitar la información a un agente sobre un OID particular. **GET NEXT / BULK** permite solicitar toda la información de una rama del árbol.
Cambiar configuración	**SET** permite cambiar la configuración de un dispositivo utilizando un OID.
Recibir alertas	**TRAP** son mensajes de alerta que envían los dispositivos administrados a una entidad administradora, son asincrónicos y requieren de una aplicación en escucha en la estación de monitoreo.

Parámetros comunes de de los mensajes	OID (identificación de objeto) *ej: 1.3.6.1.2.1.1.5.0* COMUNIDAD (identificación de “usuario”) *ej public, private (v1, v2)* TIPO DE DATO (para los mensajes SET) *ej integer, bool*

TRAPS

Tiene como objeto emitir una alerta al administrador ante un evento sucedido en el dispositivo.

Originalmente seis tipos de trap estandarizados y se ha reservado un espacio para poder definir nuevos traps (propietarias).



0-Cold start: El agente ha sido inicializado/reinicializado
1-Warm start: La configuración del agente ha cambiado
2-Link down: Una interfaz se encuentra fuera de servicio
3-Link up: Una interfaz se encuentra en servicio
4-Authentication failure: Se ha recibido un requerimiento de un NMS no autorizado
5-EGP neighbor loss: Cuando un equipo adyacente se encuentra fuera de servicio
6-Enterprise: Nuevos traps incluidos por los vendedores

Resumiendo

Estación de monitoreo

Entidad administrable (Ej. switch)
Aplicación en escucha UDP 161

Get/walk OID

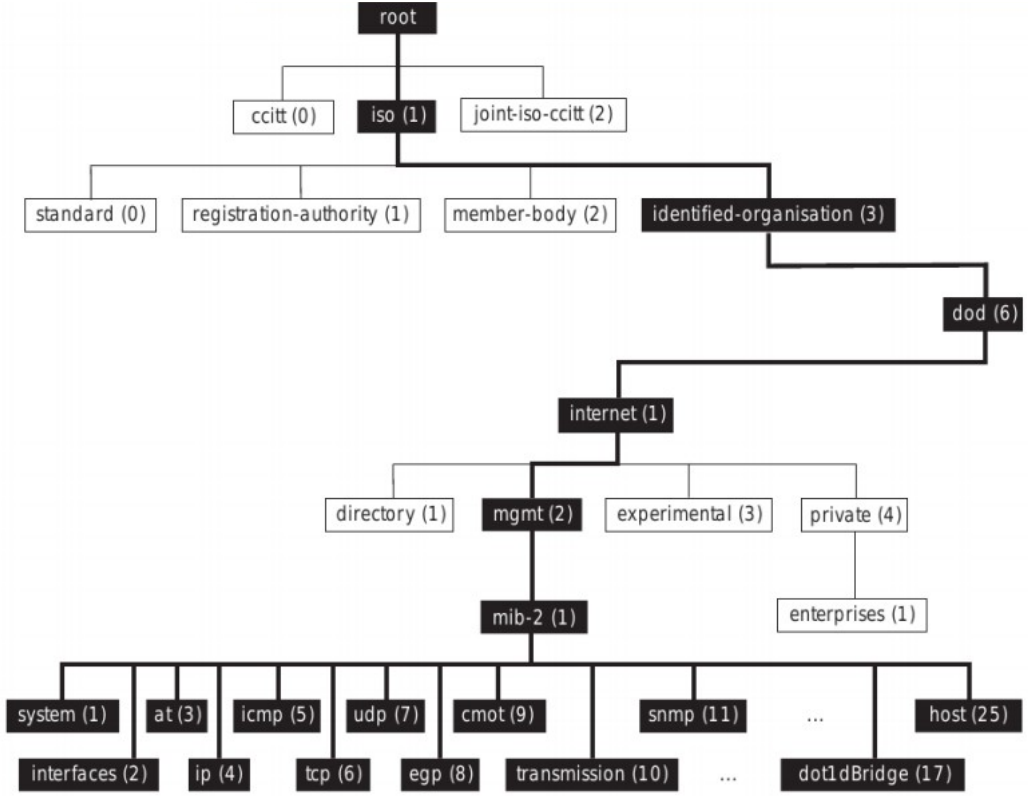
Set OID valor

TRAP

Aplicación en escucha UDP 162

OID	Descripción	Contenido
1.3.6.1.2.1.1.5.0	sysName	swtA
1.3.6.1.2.1.2.1.0	ifNumber	5
1.3.6.1.2.1.5.1.0	icmpInMsgs	10
etc	..	...

¿Cómo está compuesta una MIB?



Cada nodo del árbol tiene un nombre (nemotécnico) y un número asignado.

Cada rama del árbol representa una categoría.

Cada hoja, un objeto que puede ser consultado (y a veces modificado)

Existen ramas obligatorias de implementar (por ejemplo system, ip, icmp).

La empresas pueden crear sus propias ramas (en 1.3.6.1.4 private). Existen RFCs que definen sintáxis y semántica.

Y un OID

El objeto sysName por lo tanto viene de ISO -> la rama ORG -> la rama DOD -> Internet -> Administración -> MIB -> sistema -> y finalmente la hoja con el valor “sysName”

Management Information Base (MIB)

Constituyen una colección de objetos identificados para la gestión, sus tipos y relaciones en una entidad gestionada.

La estructura de las MIBs se establece mediante una “sintaxis” denominada Structure of Management Information (SMI) que define las reglas generales para nombrar los objetos, definir sus tipos y codificar sus valores.

Ejemplo de definición de un objeto en una MIB

{iso(1) identified-org(3) dod(6) internet(1) mgmt(2) mib-2(1) ip(4)}

ipForwarding OBJECT-TYPE
SYNTAX INTEGER {
forwarding(1), -- acting as a gateway
not-forwarding(2) -- NOT acting as a gateway
}
ACCESS **read-write**
STATUS mandatory
DESCRIPTION
"The indication of whether this entity is acting
as an IP gateway in respect to the forwarding of
datagrams received by, but not addressed to, this
entity. IP gateways forward datagrams. IP hosts
do not (except those source-routed via the host)."
::= **{ ip 1 }**

Este bloque define el OID **1.3.6.1.2.1.4.1** (ipForwarding), que permite conocer o modificar si el dispositivo gestionado reenvía paquetes o no. En palabras sencillas: si tiene rol de router (valor 1) o no (valor 2).

Management Information Base (MIB)

Definición de objetos mediante SMIv2:

```
<name> OBJECT-TYPE
SYNTAX <datatype>
UnitsParts (Opcional)
MAX-ACCESS <not-accessible, accessible-for-notify, read-only, read-write, o
read-create>
STATUS <current, obsolete, deprecated>
DESCRIPTION <Texto que describe el objeto>
ReferPart (Opcional)
IndexPart (Opcional)
DefValPart (Opcional)
::= { <OID único del objeto> }
```

Datatype puede se un tipo básico: Integer, Octect string, object identifier, Integer32, IpAddress, Counter32, Gauge32, Unsigned32, TimeTicks, Opaque, Counter64, bits;
o bien una secuencia (“sequence of”) para representar tablas

Más ejemplos de definiciones

{iso(1) ident-org(3) dod(6) internet(1) mgmt(2) mib-2(1) ip(4)}

ipDefaultTTL OBJECT-TYPE
SYNTAX **INTEGER (1..255)**
MAX-ACCESS **read-write**
STATUS current
DESCRIPTION
"The default value inserted into the Time-To-Live field of the IP header of datagrams originated at this entity, whenever a TTL value is not supplied by the transport layer protocol."
::= { ip 2 }

Este bloque define el OID **1.3.6.1.2.1.4.2** (ipDefaultTTL), que permite conocer o modificar el TTL asignado por defecto a los paquetes IP que se generan en este dispositivo.

{iso(1) ident-org(3) dod(6) internet(1) mgmt(2) mib-2(1) ip(4)}

ipInReceives OBJECT-TYPE
SYNTAX **Counter32**
MAX-ACCESS **read-only**
STATUS current
DESCRIPTION
"The total number of input datagrams received from interfaces, including those received in error."
::= { ip 3 }

Este bloque define el OID **1.3.6.1.2.1.4.3** (ipInReceives), que permite conocer la cantidad de paquetes IP recibidos por este dispositivo desde su reinicio. Como es de tipo Counter, es un valor que no puede disminuir (salvo reestablecerse a 0) y no puede ser modificado.

Management Information Base (MIB)

Algunos subconjuntos (módulos) han sido definidas por la IANA, otras por la IETF, otras por IEEE. Cada fabricante puede, además, definir las suyas.

- MIB-II: RFC 1213 — MIB for Network Management of TCP/IP-based internets
- SNMPv2-MIB: RFC 3418 — MIB for the Simple Network Management Protocol
- TCP-MIB: RFC 4022 — MIB for the Transmission Control Protocol (TCP)
- IP-MIB: RFC 4293 — MIB for the Internet Protocol (IP)
- IF-MIB: RFC 2863 — The Interfaces Group MIB
- UPS-MIB: RFC 1628 — UPS Management Information Base - IETF Tools

El sitio web OIDView [http://www.oidview.com/mibs/detail.html] posee un buscador de MIBs, donde se ordenan por fabricante y es posible conocer los OIDs de los campos de cada módulo.

OIDs clásicos

En la rama **iso.org.dod.internet.mgmt.mib-2** {1.3.6.1.2.1}

- **system** {1.3.6.1.2.1.1} -- rama de objetos de información del sistema
  - **sysDescr** {1.3.6.1.2.1.1} -- descripción textual del dispositivo
  - **sysUpTime** {1.3.6.1.2.1.3} -- tiempo desde el reinicio del dispositivo (en 1/100 seg)
  - **sysContact** {1.3.6.1.2.1.4} -- contacto de la persona que gestiona este nodo
  - **sysName** {1.3.6.1.2.1.5} -- nombre del dispositivo (hostname)

En la rama **iso.org.dod.internet.mgmt.mib-2.interfaces** {1.3.6.1.2.1.2}

- **ifTable** {1.3.6.1.2.1.2.2} -- tabla de información de las interfaces de red del dispositivo
- **ifPhysAddress** {1.3.6.1.2.1.2.2.***N***.6} -- dirección física (MAC) de la interfaz número ***N***
- **ifAdminStatus** {1.3.6.1.2.1.2.2.***N***.7} -- estado deseado de la interfaz ***N*** (up - down - testing)
- **ifOperStatus** {1.3.6.1.2.1.2.2.***N***.8} -- estado real de la interfaz número ***N*** (up - down - testing)

En la rama **iso.org.dod.internet.mgmt.mib-2.ip** {1.3.6.1.2.1.4}

- **ipInReceives** {1.3.6.1.2.1.4.3} -- número total de paquetes recibidos en todas las interfaces
- **ipForwDatagrams** {1.3.6.1.2.1.4.6} -- número total de paquetes reenviados
- **ipAddrTable.*N*.ipAdEntAddr** {1.3.6.1.2.1.4.20.***N***.1} -- rama de dirección IP de la interfaz ***N***

Ejemplo

```
:$ snmpget -v 2c -c public 54.242.136.20 1.3.6.1.2.1.1.5.0
iso.3.6.1.2.1.1.5.0 = STRING: “cisco-7513”
```

`-v 2c` Versión del protocolo a utilizar.
`-c public` la comunidad
`54.242.136.20` IP del dispositivo
`1.3.6.1.2.1.1.5.0` OID a consultar (sysName)

`iso.3.6.1.2.1.1.5.0` OID de respuesta
`STRING` tipo de dato
`“cisco-7513”` valor de la respuesta

La **comunidad** viaja en texto plano y funciona como usuario y contraseña a la vez.

Cada comunidad puede tener permisos de lectura y escritura sobre cada OID.

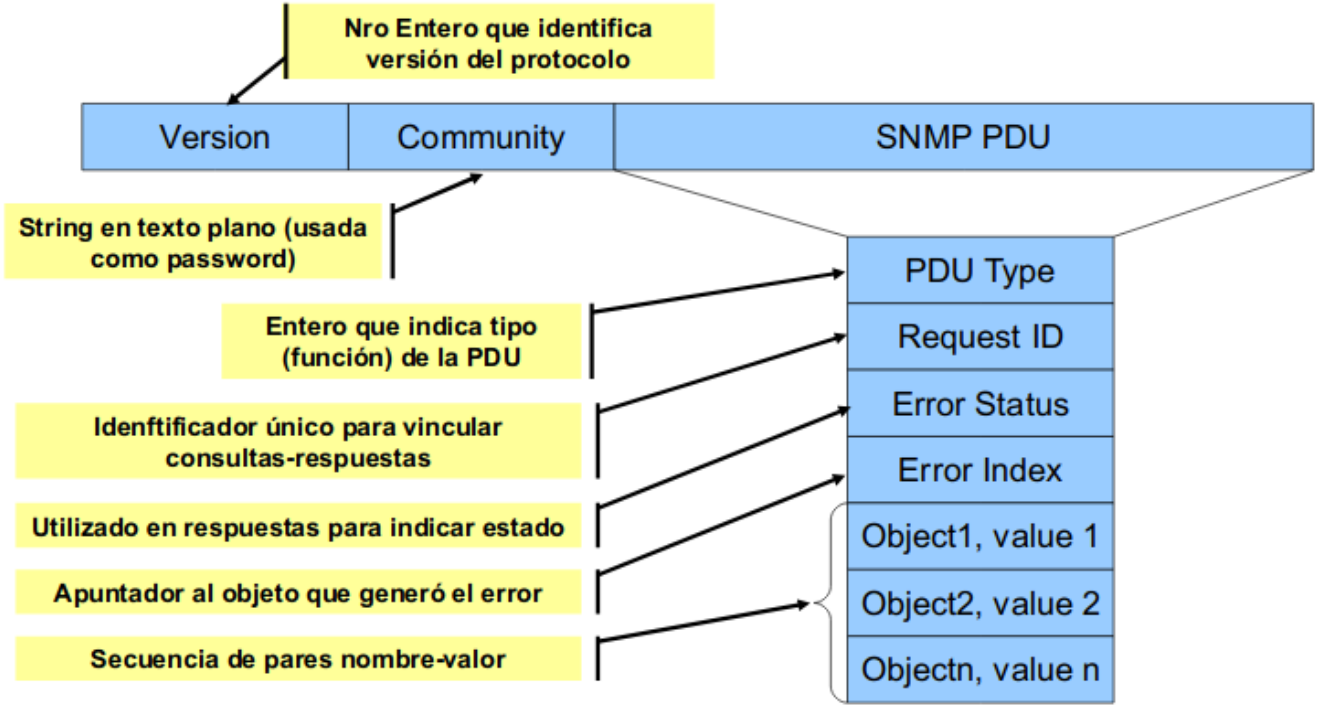
Suelen existir 2: “public” y “private”

Todos los mensajes del protocolo

Operación	Función
get-request	Solicita el valor de una variable específica
get-next-request	Solicita el valor de una variable sin especificar su nombre. Utilizada para navegar un subárbol en órden lexicográfico
set-request	Solicita la modificación del valor de una variable
get-bulk-request (v2 y 3)	Solicita un conjunto de valores en una sola operación
get-response	Respuesta a una petición get-request, get-next-request, set-request.
Inform (v2 y 3)	Permite el envío de traps con confirmación
report (v3)	Permite la comunicación entre administradores
trap	Mensaje asíncrono enviado por los agentes a la NMS para informar alguna condición especial

PDU

Formato Mensajes (Get, Get-next, Set, Response, Inform, snmpv2 trap)



SNMP V3

La versión 3 del protocolo se enfoca en proveer principalmente los siguientes servicios de seguridad:

- **Autenticación.**
- **Confidencialidad**.
- **Integridad.**

Utiliza marcas de tiempo, cifrado, ACL, HASH, y el concepto de flujos para evitar:

- Alteraciones accidentales o intencionales de los mensajes en tránsito
- Ataques por repetición (reenvío de mensajes)
- *Sniffing*
- Modificación accidental o malintencionada de información crítica.

Adaptable

SNMPv3 tiene tres opciones de configuración que puede ser especificada transacción por transacción:

- Sin autenticación y sin privacidad (*noAuthNoPriv*) - usualmente para monitoreo
- Autenticación y sin privacidad (*AuthNoPriv*) - usualmente para control
- Autenticación y privacidad (*authPriv*) - usualmente para información sensible

El protocolo SNMPv3 es modular, lo que puede permitir utilizar diferentes protocolos de monitoreo (aunque en la práctica se usa SNMPv2)

- Lenguaje de definición de datos
- Definición de información de administración (MIB)
- Un protocolo de administración
- Administración y Seguridad.

PDU Versión 3

Version	ID	Max Size	Flags	Security Model
Authoritative EngineID	Authoritative EngineBoots	Authoritative EngineTime	User Name	Security Parameters
Context Engine ID	Context name	PDU		

Cifrado si se aplica

Formato igual a SNMPv2

RMON

Se define una MIB por la IETF que permite que los dispositivos administrados puedan generar estadísticas localmente y enviar alertas definidas en base a ellas.

Para lograr esto se definen en la MIB dos tablas:

- **Tabla de control:** describe la configuración del monitor RMON especificando la información que captura y cómo lo hará
- **Tabla de datos:** almacena la información recogida que puede ser consultada

¿Que ventaja y que desventaja tiene utilizar RMON en un dispositivo?

RMON

El MIB RMON1 (el OID es 1.3.6.1.2.1.16 - iso.org.dod.internet.mgmt.mib-2.rmon) original se divide en nueve grupos:

- **Estadísticas**: Cantidad de bytes, paquetes, errores...(contadores)
- **Historia**: Muestras periódicas del grupo estadisticas.
- **Alarma**: Intervalo de muestreo y umbral de alarma para cualquier dato grabado por el agente RMON.
- **Hosts**: Datos sobre cada nodo de la red detectado.
- **HostTopN**: Lista ordenada por parámetros de objetos creados a partir de la tabla "host".
- **Matriz**: Información sobre tráfico y errores entre dos nodos.
- **Filtro** : Observar paquetes que “matchean” con un filtro. (patrón o estado)
- **Captura de paquetes**: Almacena paquetes que matchean con un filtro.
- **Evento**: Controla los eventos generados por alarmas.

RMON2 MIB agrega diez grupos más….

Bibliografía

- MAURO, D., SCHMIDT, K. Essential SNMP (2da Ed.), Douglas R. Mauro and Kevin J. Schmidt. O'Reilly Media, 2005
  - Capítulo 1: "Introduction to SNMP and Network Management"
  - Capítulo 2: "SNMPv1 and SNMPv2"
  - Capítulo 3: "SNMPv3"
- GORALSKI, W. 2017. The Illustrated Network: How TCP/IP Works in a Modern Network (2nd ed). Morgan Kaufmann.
  - Capítulo 28: "Simple Network Management Protocol (SNMP)"
- http://www.snmp.com/snmpv3/snmpv3_intro.shtml consultado en 20 de marzo 2020
- RFC's 2578, 1213, 3411-3418, 5590