

# SNMP

Area de Ingeniería Telemática  
<http://www.tlm.unavarra.es>

Grado en Ingeniería en Tecnologías de  
Telecomunicación, 4º

# SNMP

- *Simple Network Management Protocol*
- Simple, en especial comparado con CMIP
- Trabaja con valores escalares
- Diseñado para gestión en Internet (redes TCP/IP)
- Evolucionado a gestionar todo tipo de equipos (por ejemplo switches WAN)
- Versión 1
  - Quedó como “*Informational*” la parte de notificación de eventos
  - Para autenticación solo define el empleo de “*community strings*”
  - Ambos aspectos eran controvertidos en su época
  - RFC 1157, hoy histórico
- Versión 2 (RFC 1448 inicialmente, ahora 3416)
  - Aumenta los tipos de datos (contadores de 64 bits)
  - Mejor rendimiento obteniendo gran cantidad de datos
  - Notificación de eventos
  - Sin consenso en seguridad (hay propuestas tipo SNMPv2u)

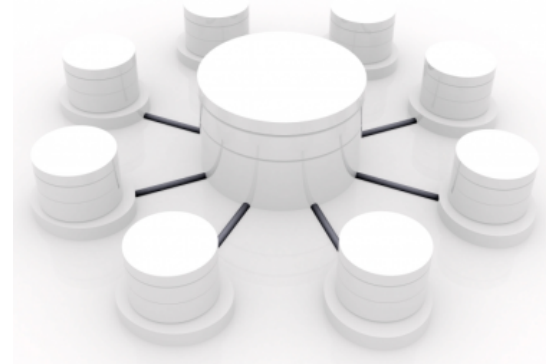
# SNMPv3

- Estandariza las propuestas de modificación a SNMPv2 para lograr seguridad y administración del protocolo con la calidad esperada por un operador
- La MIB es la MIB-2 de SNMPv2
- El protocolo y su transporte es el de SNMPv2:
  - RFC 3416 “Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)”
- Hablaremos principalmente de SNMPv2 pues dejamos la seguridad para otra asignatura
- Al no ser seguro SNMPv2 muchos fabricantes no han implementado la posibilidad de modificar los valores de la MIB por SNMP



# SNMPv2: el protocolo

- Tres tipos de operaciones:
  - “Get”: Obtener el valor de objetos de la MIB de un agente
  - “Set”: Modificar el valor de un objeto de la MIB de un agente (o crear filas en una tabla)
  - “Trap”: Notificaciones asíncronas que envía el agente al gestor
- Soporta gestión centralizada o distribuida
  - Distribuido permite mayor escalabilidad
  - Gestor gestiona a gestores intermedios
  - El gestor intermedio puede dar resúmenes de los agentes y traps a su gestor



# SNMPv2: Políticas de acceso

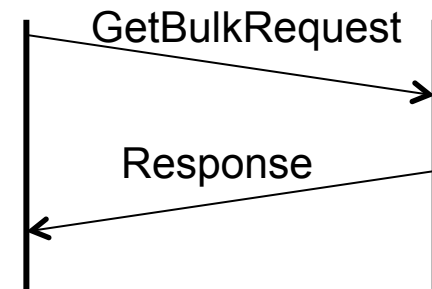
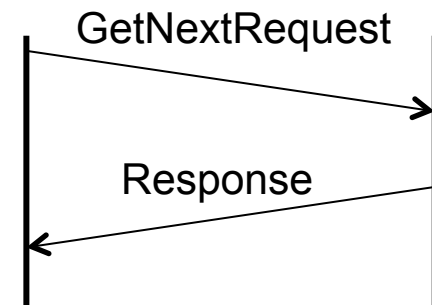
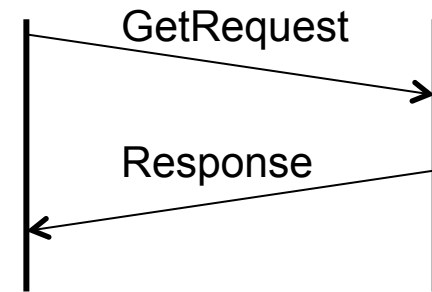
- La autenticación
  - La que ofrece SNMPv1, que es en base a nombre de “comunidad”
  - Una comunidad SNMP es una relación entre un agente y unos gestores que define autenticación y control de acceso (y proxy)
  - El nombre de la comunidad se emplea en los mensajes
  - Si conoces el nombre de comunidad puedes gestionar el agente
  - “Community based SNMPv2” o “SNMPv2c”
- Control de acceso
  - Se pueden definir diferentes niveles de acceso según la comunidad
  - Se habla de un “*community profile*”
  - Se da acceso a una “*SNMP MIB view*”, una vista de la MIB
  - Y un nivel de acceso (read-only o read-write)



# Mensajes

## GetRequest, GetNextRequest, GetBulkRequest

- Para obtener valores
- *GetRequest* puede solicitar uno o varios valores
- *GetNextRequest* permite recorrer lexicográficamente la MIB
- *GetBulkRequest* es una forma más sencilla de pedir un gran bloque de datos en una sola petición
- *Response* como mensaje de respuesta



# Mensajes

## SetRequest

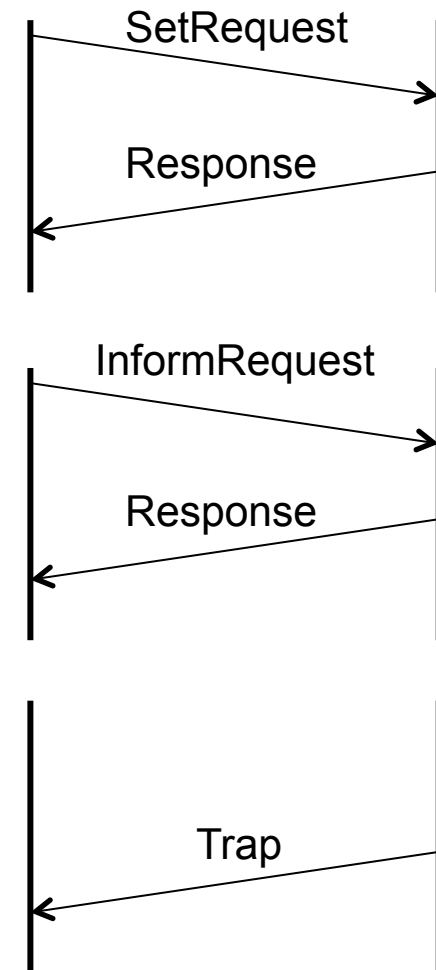
- Para asignar valor a uno o varios objetos de una MIB

## InformRequest

- Usado entre gestores
- Uno informa a otro de datos

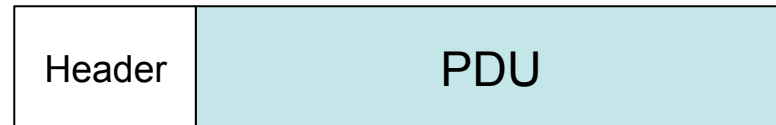
## Trap

- En respuesta a un suceso
- Ej: enlace se activa o desactiva, pérdida de vecindad en un protocolo, un fallo de autenticación, etc.



# Paquete SNMP

- Sobre UDP
- Cabecera
  - Versión (0=SNMPv1, 1=SNMPv2)
  - Comunidad
- PDU según el tipo
- ¿Cómo codificar el ASN.1?
  - BER = Basic Encoding Rules
  - Es tipo TLV (Type-Length-Value)



PDU type	Request-ID	0	0	Variable bindings
----------	------------	---	---	-------------------

(a) GetRequest-PDU, GetNextRequest-PDU, SetRequest-PDU, SNMPv2-Trap-PDU, InformRequest-PDU

PDU type	Request-ID	Error-status	Error index	Variable bindings
----------	------------	--------------	-------------	-------------------

(b) Response-PDU

PDU type	Request-ID	Nonrepeaters	Max-repetitions	Variable bindings
----------	------------	--------------	-----------------	-------------------

(c) GetBulkRequest-PDU

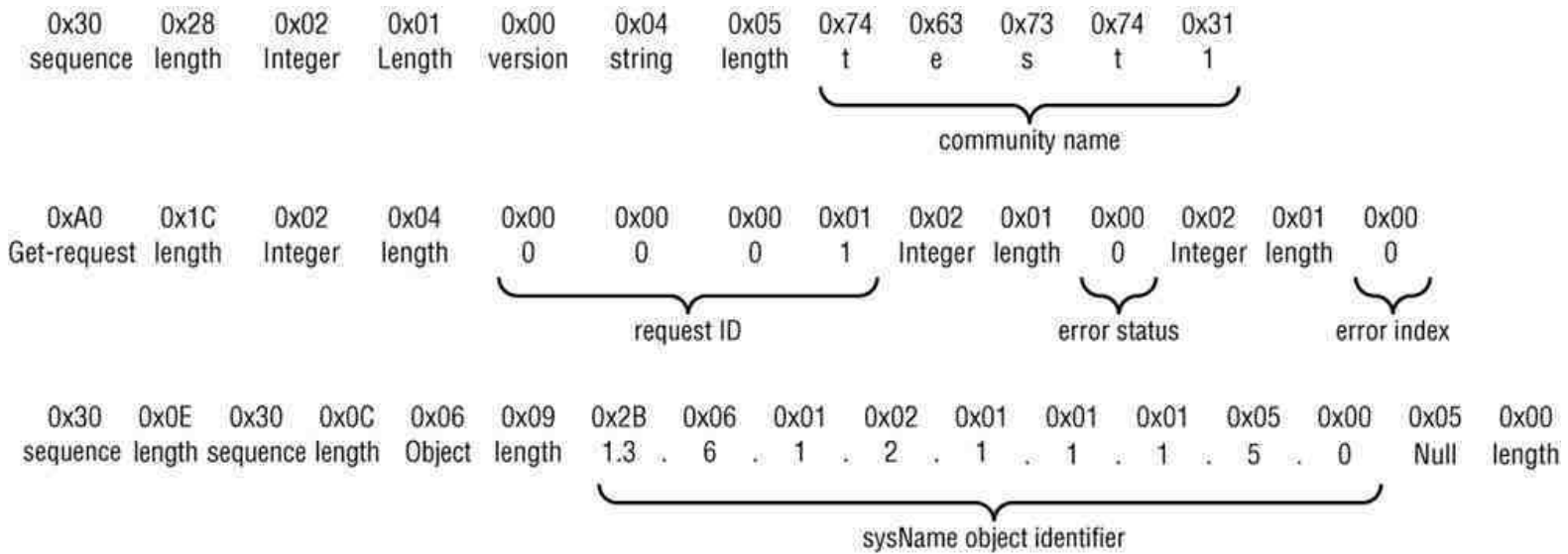
name1	value1	name2	value2	...	namen	valuen
-------	--------	-------	--------	-----	-------	--------

(d) Variable bindings



# Ejemplo

- Paquete con PDU GetRequest



# Transporte de SNMP

- RFC 3417 “Transport Mappings for the Simple Network Management Protocol (SNMP)”
- Sobre UDP sobre IPv4
  - Serialización mediante BER
  - Agente espera en puerto 161
  - Gestor espera traps en puerto 162
- Sobre OSI (opcional)
- Sobre DDP (Datagram Delivery Protocol, AppleTalk) (opt.)
- Sobre IPX (opcional)
- Sobre IEEE 802.3 (opcional)
  - RFC 4789 “Simple Network Management Protocol (SNMP) over IEEE 802 Networks”
  - Ethertype 0x814c

# Trabajo

- Objetivos:
  - Instalación y prueba de un software de gestión
  - Preferiblemente:
    - Que sea gratuito o con suficiente licencia de demostración para probarlo
    - Que se pueda instalar en GNU/Linux
  - Creación de una *testbed*
- Entregables:
  - Demostración en vivo explicando las funcionalidades que ofrece, cómo se instala, configura, etc
- Evaluación:
  - Profundidad de estudio de las capacidades del soft
  - Heterogeneidad en la *testbed*
  - Claridad de exposición y respuesta a preguntas
  - Tras 4 sesiones de laboratorio, es decir, si no intercalamos teoría sería el miércoles 2 de octubre

# Trabajo: ejemplos

- OpenNMS
  - <http://www.opennms.org>
  - <http://demo.opennms.org/opennms/index.jsp>
- JFFNMS
  - <http://www.jffnms.org>
  - <http://www.jffnms.org/screenshots/>
- NetXMS
  - <http://www.netxms.org>
  - <http://www.netxms.org/screenshots/>
- PandoraFMS
  - <http://pandorafms.com>
  - [http://firefly.artica.es/pandora\\_demo/](http://firefly.artica.es/pandora_demo/)
- OPMANTEK
  - <https://opmantek.com>
- GroundWork
  - <http://www.gwos.com>
  - <http://www.gwos.com/resources/screenshots/>

# Trabajo: ejemplos

- Icinga
  - <https://www.icinga.org>
  - <http://web.demo.icinga.org/icinga-web/>
- Cacti
  - <http://cacti.net>
- Netdisco
  - <http://www.netdisco.org>
- Zabbix
  - <http://www.zabbix.com>
  - <http://www.zabbix.com/screenshots.php>