

Tema 3. SNMP v2

1. SNMPv2.

- 1.1. Introducción.
- 1.2. Características generales.
- 1.3. Estructura de la Información de Gestión.
- 1.4. Protocolo.

Bibliografía.

Barba Martí, A., “**Gestión de red**”, Edicions UPC, 1999.

Stallings, W. “**SNMP, SNMPv2, SNMPv3, and RMON 1 and 2**” 3^a Edición, Addison Wesley, 1999.

Request For Comments (RFC)

Tema 3. SNMP v2

1. SMNPv2.

1.1. Introducción.

- ◆ Surge ante la necesidad de cubrir algunas de las deficiencias del SNMP original (que se denomina SNMPv1 a partir de la definición de SNMPv2).
- ◆ Se compone por la unión de dos iniciativas:
 - ◆ Mejorar la seguridad de SNMPv1: “Secure SNMP”.
 - ◆ Nuevos aspectos de gestión (funcionalidad, eficiencia de operación, rendimiento...): Simple Management Protocol (SMP).

Ambas se terminan a mediados de 1.992.

- ◆ SNMPv2 fue desarrollado durante finales 1.992 y presentado en marzo de 1.993. (Descrito inicialmente en las RFCs 1441-1452, actualmente obsoletas)

Tema 3. SNMP v2

1. SMNPv2.

1.1. Introducción.

- ◆ SNMPv2 fue modificado en 1.996. Se eliminaron las medidas de seguridad por problemas en los puntos de vista en el desarrollo y por la necesidad de terminar el proceso, lo cual propició el desarrollo de la versión 3 (SNMPv3).
- ◆ SNMPv2, a pesar de su mayor complejidad, es una evolución natural de SNMPv1, pero esta mayor complicación lo hace más difícil de implementar, y por tanto mas difícil que su uso se extienda rápidamente.
- ◆ Adicionalmente, para cubrir los mismos aspectos que intentaba abarcar SMP, y mejorar SNMP, aparece RMON (Remote Network Management).

Tema 3. SNMP v2

1. SMNPv2.

1.1. Introducción.

Documentos que cubren la definición de SNMPv2 de 1.996 (RFCs):

- ◆ 1901 Introduction to Community-Based SNMPv2
- ◆ 1902 Structure of Management Information for SNMPv2 (obsoleta por la RFC 2578)
- ◆ 1903 Textual Conventions for SNMPv2 (obsoleta por la RFC 2579)
- ◆ 1904 Conformance Statements for SNMPv2 (obsoleta por la RFC 2580)
- ◆ 1905 Protocol Operations for SNMPv2 (obsoleta por la RFC 3416)
- ◆ 1906 Transport Mappings for SNMPv2 (obsoleta por la RFC 3417)
- ◆ 1907 Management Information Base for SNMPv2 (obsoleta por la RFC 3418)
- ◆ 1908 Coexistence Between Version 1 and Version 2 of the Internet-Standard Network Management Framework (obsoleta por la RFC 2576)

RFCs 2576, 2578, 2579 y 2580 redactadas en 1.999 y la 3416, 3417 y 3418 en 2.002

Tema 3. SNMP v2

1. SMNPv2.

1.2. Características generales.

- ◆ Extiende el modelo de comunicaciones considerablemente, posibilitando tanto una gestión altamente centralizada, como distribuida.
- ◆ Capacidad de los sistemas de operar tanto como agentes como gestores.
- ◆ Capacidad de comunicación gestor-gestor con la posibilidad de jerarquizar la gestión.
- ◆ Mayor eficiencia en la transferencia de la información.
- ◆ Soporta una señalización extendida de errores.
- ◆ Permite el uso de varios servicios de transporte.

Tema 3. SNMP v2

1. SMNPv2.

1.2. Características generales.

Las características y mejoras anteriores se agrupan en tres categorías claves:

- ◆ Estructura de la Información de Gestión (SMI).
- ◆ Capacidad de interacción Gestor-Gestor.
- ◆ Operaciones de protocolo.

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

La Estructura de la Información de Gestión de SNMPv2 es un super-conjunto de la de SNMPv1, con especificaciones y documentación más elaborada de los objetos gestionados y MIBs.

Principales mejoras:

- ◆ Nuevas definiciones de objetos y nueva macro OBJECT-TYPE.
- ◆ Definiciones y funciones para el manejo específico de tablas.
- ◆ Definición de una macro para las notificaciones, que incluye información adicional de referencia a otros elementos de la MIB (NOTIFICATION-TYPE macro).
- ◆ Módulos de información.

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Nuevas definiciones de objetos.

TIPO DE DATO	SNMPv1	SNMPv2	DESCRIPCIÓN
INTEGER	●	●	Como en SNMPv1
Unsigned32		●	Enteros no negativos de 0 a $2^{32}-1$ (usado para Gauge32)
Counter32	●	●	Como en SNMPv1
Counter64		●	Como Counter32 pero de 0 a $2^{64}-1$
Gauge32	●	●	Como en SNMPv1
TimeTicks	●	●	Como en SNMPv1
OCTET STRING	●	●	Tamaño de 0 a 65535 caracteres
IpAddress	●	●	Como en SNMPv1
OBJECT IDENTIFIER	●	●	Como en SNMPv1
Opaque	●	●	Se mantiene en SNMPv2 por compatibilidad con SNMPv1
BITS		●	Lista enumerada de bits identificados con nombre

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Modificaciones y Nuevas cláusulas y especificaciones a la macro OBJECT-TYPE.

- ◆ **UnitsPart**. Es una cláusula opcional que permite definir de forma textual, qué unidades están asociadas con el objeto.

- ◆ **MAX-ACCESS**. Cambian los tipos de acceso a un objeto. En orden de menor a mayor capacidad:
 - ◆ **not-accessible**: El gestor no puede acceder al objeto para ninguna operación.
 - ◆ **accessible-for-notify**: Sólo accesible vía una notificación.
 - ◆ **read-only**: Acceso solamente para lectura.
 - ◆ **read-write**: Permisos para leer y escribir el objeto.
 - ◆ **read-create**: Permiso para leer, escribir y para crear una instancia del mismo.

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Modificaciones y Nuevas cláusulas y especificaciones a la macro OBJECT-TYPE.

- ◆ **STATUS.** Desaparecen mandatory y optional, y aparecen nuevas categorías:
 - ◆ **current:** El objeto es válido para el estándar actual.
 - ◆ **obsolete:** El objeto no debería ser implementado.
 - ◆ **deprecated:** El objeto está obsoleto pero puede ser necesario para la compatibilidad con aplicaciones antiguas.

- ◆ **ReferPart.** Cláusula textual opcional que define una referencia cruzada con otro módulo de la MIB.

- ◆ **IndexPart.** Es más compleja que en SNMPv1, ya que se usa para el manejo de las tablas en SNMPv2.

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Modificaciones y Nuevas cláusulas y especificaciones a la macro OBJECT-TYPE.

- DefValPart.** Es opcional e indica una valor por defecto aceptable para cuando se crea una instancia del objeto. Se deja a discreción de agente.

- VALUE NOTATION.** El nombre usado por SNMPv2 para acceder a este objeto vía SNMPv2.

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Macro OBJECT-TYPE (*RFC 2578 SMIv2, página 8*).

```
OBJECT-TYPE MACRO ::=
BEGIN
  TYPE NOTATION ::=
    "SYNTAX" Syntax
    UnitsPart
    "MAX-ACCESS" Access
    "STATUS" Status
    "DESCRIPTION" Text
    ReferPart
    IndexPart
    DefValPart

  VALUE NOTATION ::= value(VALUE ObjectName)
  Syntax ::= type | "BITS" "{" NamedBits "}"
  NamedBits ::= NamedBit | NamedBits "," NamedBit
  NamedBit ::= identifier "(" number ")"
```

Continúa →

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Macro OBJECT-TYPE (*RFC 2578 SMIV2, página 8*).

```
UnitsPart ::= "UNITS" Text | empty
Access ::= "not-accessible" | "accessible-for-notify" | "read-only" | "read-write" | "read-create"
Status ::= "current" | "deprecated" | "obsolete"
ReferPart ::= "REFERENCE" Text | empty
IndexPart ::= "INDEX" "{" IndexTypes "}" | "AUGMENTS" "{" Entry "}" | empty
IndexTypes ::= IndexType | IndexTypes "," IndexType
IndexType ::= "IMPLIED" Index | Index
Index ::= value(ObjectName)
Entry ::= value(ObjectName)
DefValPart ::= "DEFVAL" "{" Defvalue "}" | empty
Defvalue ::= value(ObjectSyntax) | "{" BitsValue "}"
BitsValue ::= BitNames | empty
BitNames ::= BitName | BitNames "," BitName
BitName ::= identifier
Text ::= value(IA5String)
```

END

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Definición de tablas conceptuales.

- Se definen dos tipos de tablas, aquellas que tan sólo puede crear o borrar filas el agente y no el gestor (como la lista de líneas físicas disponibles), y aquellas que sí puede modificar el gestor, como una tabla de direcciones de enrutamiento.
- Funcionan con un sistema de indexación por tipo e índice (cláusula **INDEX**).
- Proporcionan una cláusula **AUGMENTS** (alternativa a **INDEX**), para permitir incrementar el número de columnas en una tabla sin tener que re-escribir su definición.

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Definición de tablas conceptuales.

Una tabla conceptual tiene la siguiente sintaxis:

SEQUENCE OF <EntryType>

Donde <EntryType> se refiere al tipo SEQUENCE de su fila conceptual subordinada.

Una fila conceptual tiene la siguiente sintaxis:

<EntryType>

Donde <EntryType> es un tipo SEQUENCE definido como sigue:

<EntryType> ::= SEQUENCE { <type1>, ... , <typeN> }

Donde hay un <type> por cada objeto subordinado, y cada <type> es de la forma:

<descriptor> <syntax>

Donde <descriptor> es el descriptor que nombra al objeto subordinado,

y <syntax> tiene el valor de la cláusula SINTAX de ese objeto subordinado.

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Macro NOTIFICATION-TYPE (*RFC 2578 SMIV2, página 10*).

```
NOTIFICATION-TYPE MACRO ::=
BEGIN
    TYPE NOTATION ::=
        ObjectsPart
        "STATUS" Status
        "DESCRIPTION" Text
        ReferPart

    VALUE NOTATION ::= value(VALUE NotificationName)
    ObjectsPart ::= "OBJECTS" "{" Objects "}" | empty
    Objects ::= Object | Objects "," Object
    Object ::= value(ObjectName)
    Status ::= "current" | "deprecated" | "obsolete"
    ReferPart ::= "REFERENCE" Text | empty
    Text ::= value(IA5String)

END
```

La macro NOTIFICATION-TYPE se usa para definir la información que envía una entidad SNMPv2 cuando ocurre un evento excepcional.

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Módulos de Información.

Existen tres tipos de módulos de información. Cada módulo comienza con la macro **MODULE-IDENTITY**, la cual da información de contacto e historial de revisiones.

- ◆ Módulos MIB: Éstos relacionan los objetos gestionados y usan las macro **OBJECT-TYPE** y **NOTIFICATION-TYPE**.
- ◆ Declaraciones de conformidad para los módulos MIB. Hacen uso de las macros **OBJECT-GROUP** y **MODULE-COMPLIANCE**.
- ◆ Declaraciones de Capacidad para la implementación de Agentes, las cuales usan las macros **AGENT-CAPABILITIES**.

Tema 3. SNMP v2

1. SMNPv2.

1.3. Estructura de la Información de Gestión.

Módulos de Información: Macro MODULE-IDENTITY.

```
MODULE-IDENTITY MACRO ::=
BEGIN
    TYPE NOTATION ::=
        "LAST-UPDATED" value(Update ExtUTCTime)
        "ORGANIZATION" Text
        "CONTACT-INFO" Text
        "DESCRIPTION" Text
        RevisionPart
    VALUE NOTATION ::= value(VALUE OBJECT IDENTIFIER)
    RevisionPart ::= Revisions | empty
    Revisions ::= Revision | Revisions Revision
    Revision ::=
        "REVISION" value(Update ExtUTCTime)
        "DESCRIPTION" Text
    Text ::= value(IA5String)
END
```

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Las PDU de SNMPv2 van encapsuladas en un mensaje como en SNMPv1.

Los mensajes de SNMPv2 proveen la funcionalidad necesaria para las características de seguridad que proporciona éste.

SNMPv2 provee tres tipos de acceso a la información de gestión:

- ◆ **GESTOR-AGENTE, pregunta-respuesta.**
- ◆ **GESTOR-GESTOR, pregunta-respuesta (nuevo con respecto SNMPv1).**
- ◆ **AGENTE-GESTOR, sin confirmar (traps).**

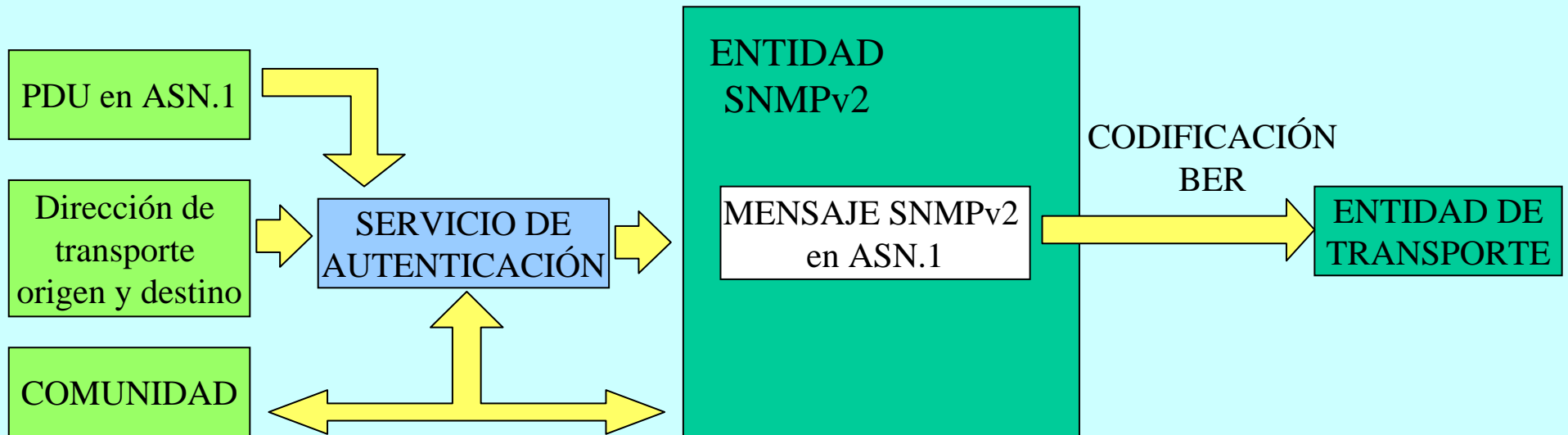
Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2.

Transmisión de un mensaje SNMPv2:



Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2.

Formato de un mensaje SNMPv2:

```
Message ::= SEQUENCE {  
    version    INTEGER { version(1) }, -- SNMPv2 la versión es 1  
    community  OCTET STRING,          -- nombre de la comunidad  
    data       ANY                     -- Una PDU SNMP  
}
```

En SNMPv2 las consideraciones hechas sobre los nombres de comunidad y los perfiles de comunidad son las mismas que para SNMPv1, debido a la falta de acuerdo en cuanto a la forma de incorporar la seguridad al protocolo en su desarrollo.

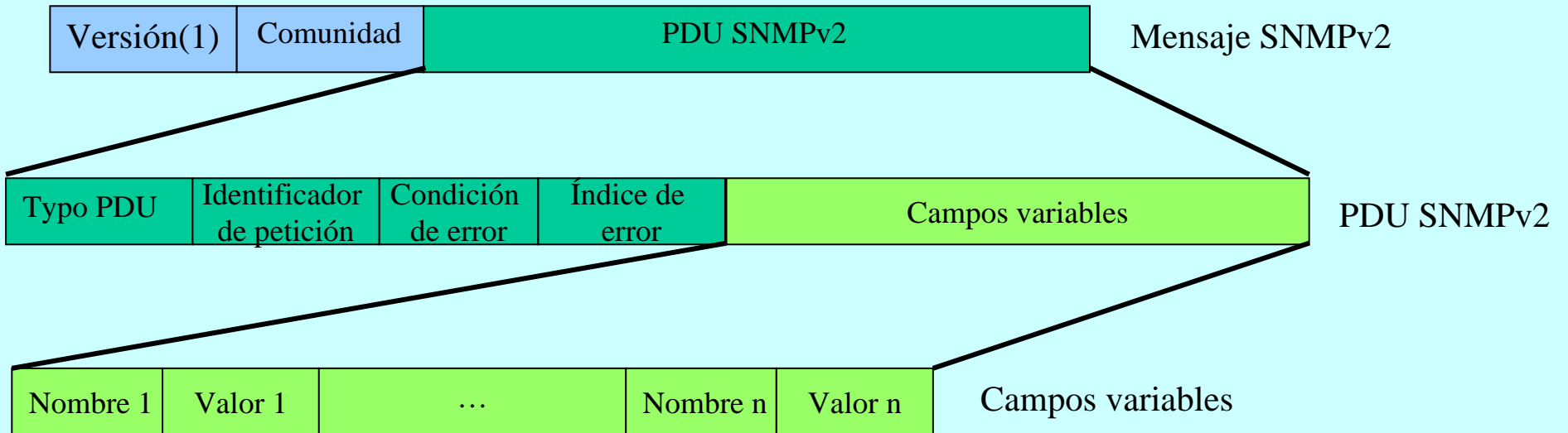
Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2.

Formato de un mensaje SNMPv2:



Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2.

PDU de SNMPv2:

Tipo PDU	Petición id	0	0	Campos variables
----------	-------------	---	---	------------------

GetRequest PDU, GetNextRequest PDU, SetRequest PDU, SNMPv2-Trap PDU, InformRequest PDU

Tipo PDU	Petición id	error-status	error-índex	Campos variables
----------	-------------	--------------	-------------	------------------

Response PDU

Tipo PDU	Petición id	non-repeates	max-repitions	Campos variables
----------	-------------	--------------	---------------	------------------

GetBulkResquest PDU

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. Codificación ASN.1 de las PDUs

```
SNMPv2-PDU DEFINITIONS ::= BEGIN

    IMPORTS
        ObjectName, ObjectSyntax, Integer32 FROM SNMPv2-SMI;

    PDUs ::=
        CHOICE {
            get-request          GetRequest-PDU,
            get-next-request     GetNextRequest-PDU,
            get-bulk-request     GetBulkRequest-PDU,
            response             Response-PDU,
            set-request          SetRequest-PDU,
            inform-request       InformRequest-PDU,
            snmpV2-trap          SNMPv2-Trap-PDU,
            report               Report-PDU,
        }
```


Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. Codificación ASN.1 de las PDUs

```
GetRequest-PDU ::= [0] IMPLICIT PDU
GetNextRequest-PDU ::= [1] IMPLICIT PDU
Response-PDU ::= [2] IMPLICIT PDU
SetRequest-PDU ::= [3] IMPLICIT PDU
-- [4] is obsolete
GetBulkRequest-PDU ::= [5] IMPLICIT BulkPDU
InformRequest-PDU ::= [6] IMPLICIT PDU
SNMPv2-Trap-PDU ::= [7] IMPLICIT PDU
-- Usage and precise semantics of Report-PDU are not presently
-- defined. Any SNMP administrative framework making use of
-- this PDU must define its usage and semantics.
Report-PDU ::= [8] IMPLICIT PDU

max-binding INTEGER ::= 2147483647
```

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. Codificación ASN.1 de las PDUs

```
PDU ::=
  SEQUENCE {
    request-id      Integer32,
    error-status    INTEGER {
      noError(0), tooBig(1),noSuchName(2),    -- for proxy compatibility
      badValue(3),    -- for proxy compatibility
      readOnly(4),    -- for proxy compatibility
      genErr(5), noAccess(6), wrongType(7), wrongLength(8),
      wrongEncoding(9), wrongValue(10), noCreation(11), inconsistentValue(12),
      resourceUnavailable(13), commitFailed(14), undoFailed(15),
      authorizationError(16), notWritable(17), inconsistentName(18)
    },
    error-index     INTEGER (0..max-bindings),
    variable-bindings  VarBindList
  }
```

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. Codificación ASN.1 de las PDUs

```
BulkPDU ::= SEQUENCE {  
    request-id      Integer32,  
    non-repeaters   INTEGER      (0..max-bindings),  
    max-repetitions INTEGER      (0..max-bindings),  
    variable-bindings -- values are ignored  
    VarBindList  
}
```

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. Codificación ASN.1 de las PDUs

```
VarBind ::= SEQUENCE {
    name      ObjectName,
    CHOICE {
        value
        ObjectSyntax,
        unspecified      -- in retrieval requests
        NULL,
        -- exceptions in responses
        noSuchObject     [0]    IMPLICIT NULL,
        noSuchInstance   [1]    IMPLICIT NULL,
        endOfMibView     [2]    IMPLICIT NULL
    }
}

VarBindList ::= SEQUENCE (SIZE (0..max-bindings)) OF  VarBind

END
```

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. Caracterización de las PDUs

SNMPv1	SNMPv2	Dirección	Descripción
GetRequest	GetRequest	Gestor → Agente	Solicita el valor de cada objeto en la lista
GetNextRequest	GetNextRequest	Gestor → Agente	Solicita el siguiente valor para cada objeto listado
-	GetBulkRequest	Gestor → Agente	Solicita múltiples valores
SetRequest	SetRequest	Gestor → Agente	Establece el valor para cada objeto listado
-	InformRequest	Gestor → Gestor	Transmite información de gestión no solicitada
GetResponse	Response	Agente → Gestor Gestor → Gestor (SNMPv2)	Responde a una petición de un gestor
Trap	SNMPv2-Trap	Agente → Gestor	Transmite información no solicitada asociada a un evento.

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. Códigos de error.

Error(código)	GetRequest	SetRequest	InformRequest
	GetNextRequest GetBulkRequest		
noError(0)	●	●	●
tooBig(1)	●	●	●
noSuchName(2)			
badValue(3)			
readOnly(4)			
genError(5)	●	●	●
noAccess(6)		●	
wrongType(7)		●	
wrongLegth(8)		●	
wrongEncoding(9)		●	

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. Códigos de error.

Error(código)	GetRequest GetNextRequest GetBulkRequest	SetRequest	InformRequest
wrongValue(10)		●	
noCreation(11)		●	
inconsistentValue(12)		●	
resourceUnavalilable(13)		●	
commitFailed(14)		●	
undoFailed(15)		●	
autorizationError(16)	●	●	●
noWritable(17)		●	
inconsistentName(18)		●	

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. Códigos de error.

Error(código)	
noError(0)	Sin error
tooBig(1)	Respuesta mas larga de lo que admite el formato de la PDU
noSuchName(2)	
badValue(3)	
readOnly(4)	
genError(5)	Error generico
noAccess(6)	la variable no es soportada por la vista del MIB para esa operación
wrongType(7)	el nuevo valor proporcionado es de un tipo de datos ASN.1 erróneo.
wrongLegth(8)	el nuevo valor proporcionado es de longitud errónea
wrongEncoding(9)	el nuevo valor proporcionado está codificado erróneamente

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. Códigos de error.

Error(código)	
wrongValue(10)	el nuevo valor está fuera del rango de valores admitidos para el tipo de objeto correspondiente
noCreation(11)	la variable no existe y el agente no puede crear instancias del tipo objeto correspondiente
inconsistentValue(12)	el valor proporcionado es inconsistente con los valores de otro objeto en el agente
resourceUnavalilable(13)	un recurso requerido no puede ser reservado
commitFailed(14)	indica que hubo un fallo en la segunda pasada pero que se deshicieron los cambios.
undoFailed(15)	indica que falló la segunda pasada y que algunos cambios no pudieron deshacerse.
autorizationError(16)	
noWritable(17)	La variable existe pero el agente no puede modificar instancias del tipo objeto correspondiente.
inconsistentName(18)	la variable no existe y no puede ser creada porque el nombre de la instancia es inconsistente con los valores de otro objeto en el agente

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. *PDU*s de SNMPv2:

PDU `GetRequest`:

El formato de la petición es idéntico a SNMPv1, pero varía la respuesta, la cual ya no es atómica.

Si algún objeto provoca un error por no coincidir el identificador de objeto con uno accesible por la petición o con una variable, el valor usado es **noSuchObject** o **noSuchInstance** respectivamente, que se comunica dentro de los campos variables junto al identificador de objeto correspondiente.

Si es otro error se usan los campos **error-status** y **error-index** como en SNMPv1.

Si la respuesta es demasiado grande por causa de una limitación de tamaño de PDU local, se genera una nueva con el error **tooBig(1)** y con el campo variable vacío.

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. *PDU*s de SNMPv2:

PDU **GetNextRequest**:

Igual en sintaxis y significado que en SNMPv1, salvo que la respuesta no es atómica. Ya que **GetNextRequest** en SNMPv2 procesa tantas variables como sea posible.

La respuesta contiene en la lista de variables el identificador de objeto y el valor en el caso de encontrar el objeto, o el valor **endOfMibView** si no hay un sucesor en orden lexicográfico.

En caso de error se procede al igual que con **GetRequest**.

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. *PDU*s de SNMPv2:

PDU **GetBulkRequest**:

Esta PDU es una de la mayores mejoras de SNMPv2, ya que permite el intercambio de grandes cantidades de información minimizando el número de peticiones.

GetBulkRequest sigue los principios de **GetNextRequest**, tiene una primera parte de peticiones de valores de objetos que es como una petición **GetNextRequest**, con tantos objetos como el valor del campo **non-repeaters**,

La segunda parte comienza a partir del objeto **non-repeaters+1**. De estos objetos se intentará devolver un número de sucesores en orden lexicográfico igual al campo **max-repetitions**.

Si no hay un sucesor el valor devuelto es **endOfMibView**. Si una variable falla por otra razón distinta a la anterior no se devuelve ningún valor y se activan los campos de error.

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. *PDU*s de SNMPv2:

PDU **SetRequest**:

Igual en sintaxis y significado que en SNMPv1, salvo en la forma de procesar la respuesta, en la que se dan do fases diferenciadas:

- ◆ Primero se validan todos los pares de variables con su valor, comprobando posibles condiciones de error.
- ◆ Segundo se realiza la modificación de los valores por los recibidos en la PDU **SetRequest**.

Al igual que en SNMPv1 la operación **SetRequest** es atómica.

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. *PDU*s de SNMPv2:

PDU **SNMPv2-Trap**:

Es generada y transmitida por una entidad SNMPv2 que actúa como agente ante la aparición de algún evento inusual, como en SNMPv1 pero con un formato distinto.

Usa el mismo formato que las demás PDUs, salvo GetBulkRequest y contiene dentro de los campos variables los siguientes objetos:

- ◆ sysUpTime.0
- ◆ snmpTrapOID.0: Parte del grupo de trap en la MIB de SNMPv2.
- ◆ Si la cláusula **OBJECT** está presente en la invocación de la macro **NOTIFICATION-TYPE**, cada variable y su valor se añaden a la lista de variables.
- ◆ Cualquier otra variable incorporada por el agente.

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. *PDU*s de SNMPv2:

PDU InformRequest:

Esta PDU la envía una entidad SNMPv2 actuando como gestor a otra entidad que actúe como gestor, en beneficio de la aplicación que usa ésta última para completar la información de gestión.

Usa el mismo formato que las PDU SNMPv2-Trap para los campos variables.

Cuando la entidad receptora lee una **InformRequest** construye una respuesta con los mismos valores de los campos que en la PDU entrante, salvo que sea demasiado grande, caso en el que se responde con una PDU con **error-status** igual a **tooBig**, **error-index** a cero y sin campos variables.

Tema 3. SNMP v2

1. SMNPv2.

1.4. Protocolo.

Mensajes SNMPv2. *PDU*s de SNMPv2:

PDU Report:

Esta PDU aparece en la RFC pero no tiene ninguna definición, tan sólo un comentario, que la destina para que pueda ser definida, tanto en si sintaxis como semántica, para su uso dentro de un marco administrativo de gestión concreto.