# SNMP – Simple Network Management Protocol

Är ett verktyg för att administrera nätverk, framtaget egentligen som ett provisorium. Introducerades 1989 av IAB och var tänkt som en standard för att administrera TCP/IP-nätverk över Ethernet. När SNMP blev populärt översattes (portades) till andra transportprotokoll, såsom IPX/SPX, AppleTalk samt andra OSI-protokoll.

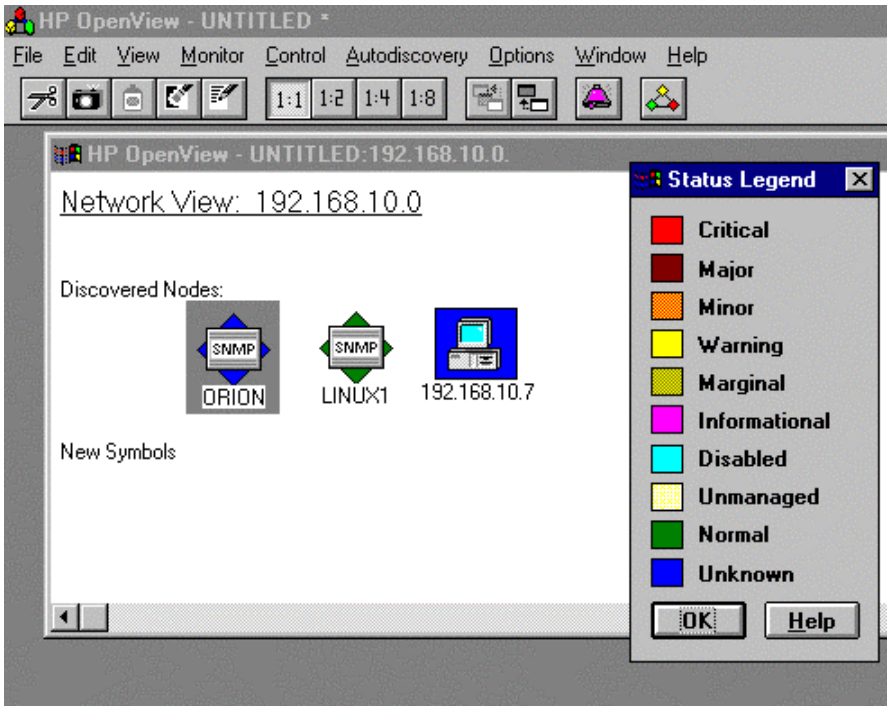
## *Nätverksövervakning*

Nätverksövervakning innebär att på ett effektivt sätt kontrollera och göra ett nätverk effektivare. ISO (International Organization for Standards) har definierat fem nyckelområden som bör hanteras av ett nätverksövervakningssystem: felhantering (Fault Management), konfigurering (Configuration Management), säkerhet (Security Management), prestanda (Performance Management) och behörighet (Accounting Management).

*Felhantering*

I felhanteringen ingår tre saker; fastställa sympton, isolera problemet och till sist föreslå en lösning. Några styrprogram visa status på SNMP-enheterna mha färgkoder enligt tabell nedan:

| Färg | Felkod |
|---|---|
| Grön | Inga fel. |
| Gul | Kan ha problem. |
| Röd | Enhet har problem och finns inte tillgänglig. |
| Blå | Enhet fungerar, men det finns något fel. |
| Orange | Konfigurations fel. |
| Grå | Finns ingen tillgänglig information om enheten. |
| Lila | Enheten har blivit förfrågad (queried) eller pollad. |



### *Konfigurering*

I ”configuration management” har du som administratör möjlighet att initierar, identifiera, konfigurera och kontrollera enheter på ditt nätverk. För att få en effektiv lösning måste inventering ske och resultatet av denna inventering skall lagras på central plats. Detta för att du skall kunna avgöra hur enheterna skall skötas eller omkonfigureras.

### *Säkerhet*

I ”Security management” sker administration av säkerheten samt accesskontroll för de olika enheterna på nätverket. Andra viktiga bitar är loggningar, krypteringar, dekrypteringar etc.

### *Prestanda*

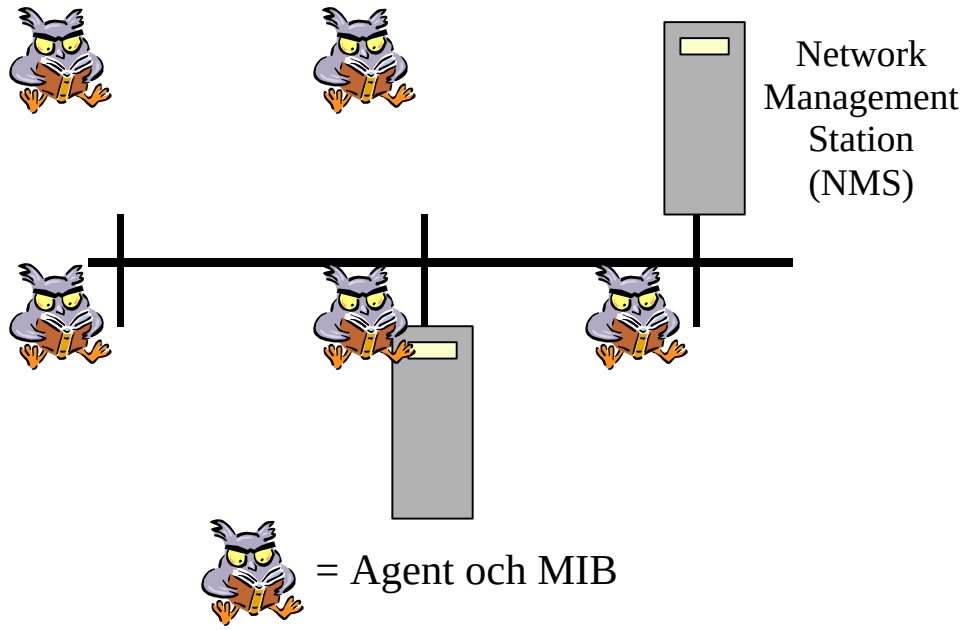
”Performance management” består av två kategorier; mätning och kontroll. Mätning innebär att logga hur användare använder nätverket. Detta sker genom olika mätpunkter ute på enheterna. Kontroll sker genom att justera olika mätpunkter. Används för att korrigera vid t ex prestanda problem.

### *Behörighet*

Behörighet eller ”Accounting management” har egentligen inte så mycket med accesskontroll eller liknande att göra. Målet med ”Accounting management” är att få ett mått på hur nätverket används, för att kunna få fram en kostandsbild. Information som fås fram kan ligga som underlag till eventuella faktureringar på hur nätverket används. Informationen som fås fram kan ligga till grund för framtida investeringar för nätverket.

### *Network Management Model*

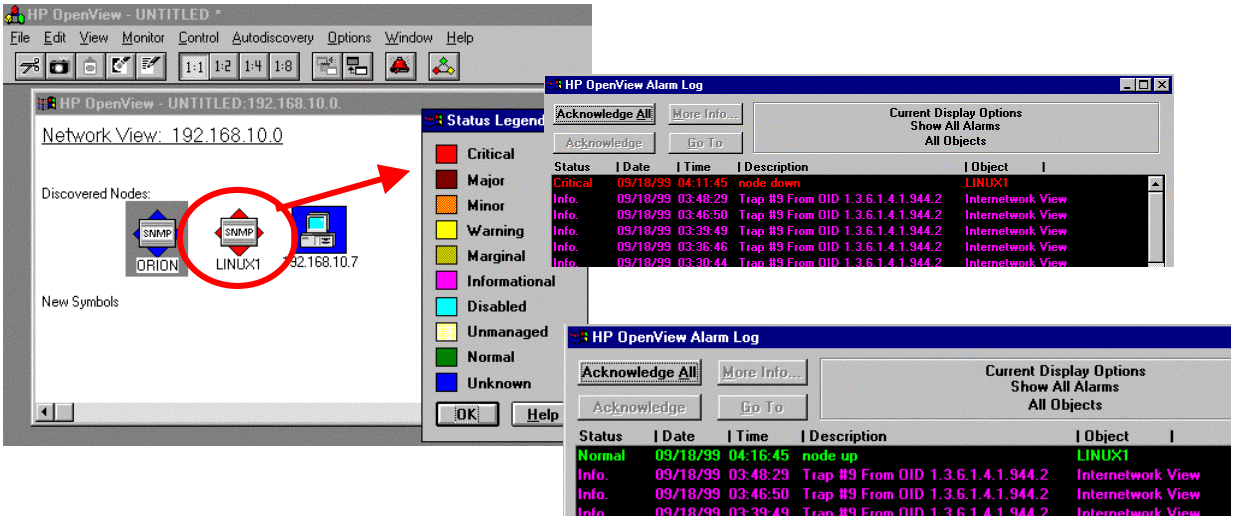
För att uppnå de fem målen i FCAPS-modellen har ISO tillsammans med skaparna av SNMP kommit överens om en modell. Denna modell består av tre element; ”managed node” (nätverksvärd med agent), informationsdatabas (SNMP Management Information Base) samt ett protokoll såsom SNMP (kan vara annat).



## *Hur fungerar det?(Översikt)*

Som vi sa tidigare så får man med hjälp av SNMP möjlighet att styra och övervaka enheter i nätverket. Aktuell status för en nod, nod kan vara repeaters, routers, gateways, brandväggar, individuella system som NT Workstation, konfigurera enheten få information och besked vid olika händelser såsom t ex avstängning av nod.

Bilderna illustrerar vad som händer i HP OpenView när jag stänger av enhet med namnet LINUX1.




Om t ex en router övervakas, kan man få information om aktuell konfiguration, routingtabeller, samt statistik om trafik och eventuella fel.

Största fördelen med SNMP är att man får en enhetlig hantering av enheter från olika tillverkare och som körs under olika operativsystem. HP OpenView som körs under NT kan ”lyssna” efter meddelande från en enhet baserad på Linux.

### *Komponenter*

Tekniken består av tre delar som arbetar tillsammans; MIB (Management Information Base), agent och ett styrprogram.

Den information som SNMP kan hämta och manipulera finns representerat i MIB. MIB är en strukturerad tabell som innehåller nätverkselement och dess attribut. Stukturen kan liknas med ett träd, nuvarande standard heter MIB-II och omfattar 171 objekttyper och finns definierat i RFC 1213.

Agent är ett program som finns ute på varje SNMP-enhet. Agentens uppgift är att samla information om sin nod (enhet).

Styrprogrammet, t ex HP OpenView (NT) eller Scotty (Linux), finns på server och har till uppgift att samla in information från agenterna. Informationen presenteras sedan på lite olika sätt beroende på hur pass avancerat styrprogrammet är. Styrprogrammet kallas för NMS-applikation och bör innehålla följande element; grafiskt användargränsnitt, säkerhet,

funktioner för insamling av information genom ”network management protocol”, snabbt kunna utvidga och anpassa programvaran, metoder för att spåra problem, databasen skall vara lätt att uppdatera och det skall finnas mekanismer för rapporter.

### *Sätt att samla in information*

De flesta övervakningsprotokoll använder sig av två sätt för att samla in information; ”traversals” och ”traps”. Andra metoder kan vara pollning eller mha proxy agents.

*Traversals* – NMS skickar förfrågan eller begäran till agenten, SNMP föredrar denna metod.

*Trap* – agent genererar en ”trap”, vilket skickas automatiskt till NMS.

*Pollning* – genom pollning kan administratören skaffa information lite snabbare. Pollning har stora likheter med trap, men den viktiga skillnaden: pollning initieras av NMS medan trap initieras av agent.

*Proxy agents* – arbetar som ombud för andra enheter.

I exemplet med LINUX-maskinen som stängdes av var HP OpenView inställt för pollning. Men både traversals och trap hade kunnat användas.

## *Vilken övervakningsstrategi*

Det finns tre möjliga arkitekturer för nätverksövervakning; centraliserad, distribuerad och hierarkisk.

*Centraliserad* – all övervakningsinformation skickas till ett system. Alla applikationer för övervakning och skötsel finns på NMS. Denna NMS svarar på alla ”trap messages” som skickas från noderna ut på nätverket. Modellen passar små till medelstora nätverk, nackdel med arkitekturen är att det blir väldigt lätt överbelastat.

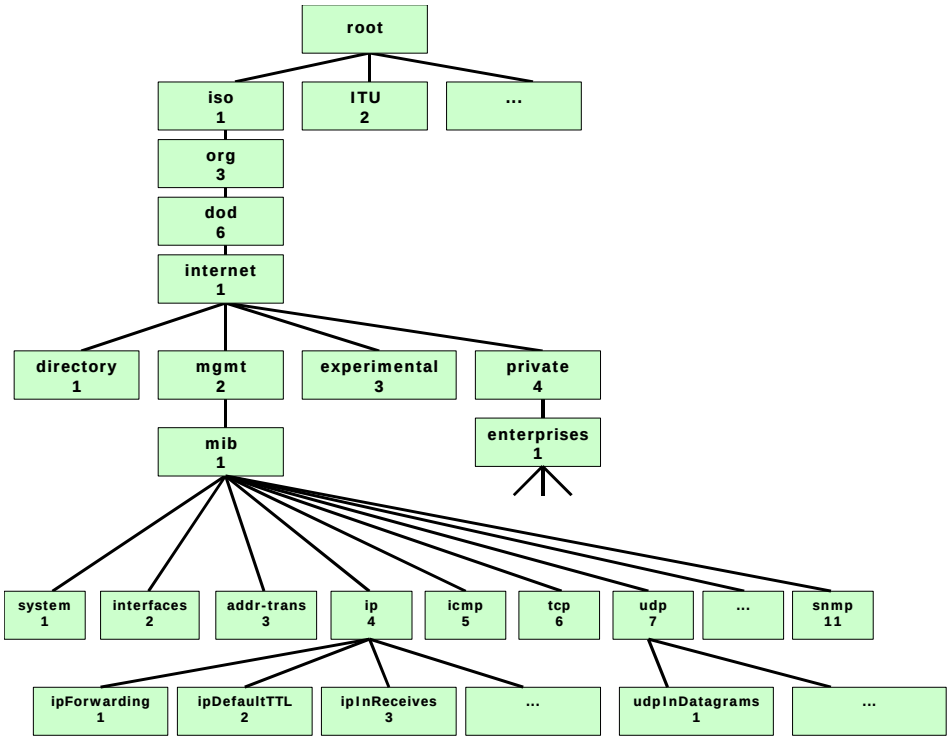
*Distribuerad* – finns ingen speciell NMS som har ansvaret för hela. NMS kan finnas utplacerat geografiskt, eller vara ansvarig för specifik typ av nätverksenhet. Passar från medel till stora nätverk, blir inte så lätt överbelastat.

*Hierarkiskt* – kombineras centraliserat system med distribuerat system. Erbjuder de fördelar som finns dels i det centraliserade typen och dels i det distribuerade.

## *MIB och objektidentifierare*

Som jag skrev tidigare, så kan MIB liknas vid ett träd, där de individuella variablarna finns allra längst ut på grenarna. Standard som är aktuell idag heter MIB-II.

Sk objektidentifierare används för att ge ett unikt namn till varje objekt i trädet. SNMP MIB innehåller tre huvudgrenar; CCITT (ITU), ISO och ISO/CCITT (ITU) (förenade).




Största delen av trädet finns under objektidentifieraren 1.3.6.1 som tillägnats Internet. Under objektet hittar vi de olika protokollen, TCP, IP, ICMP, UDP, SNMP etc, samt system och interface.

Lövobjekten kan identifieras på två sätt; men dess nummerkod eller med de olika objektnamnen. Vill jag åt ip under mib blir nummerkoden 1.3.6.1.2.1.4 eller iso.org.dod.internet.mgmt.mib.ip .

Under grenen enterprise är det öppet för tillverkare av datorutrustning att lägga till egna definitioner. IBM, Sun och HP t ex har sina egna MIB:ar, även nätverksadministratör kan göra egna tillägg i MIB:en. Det är viktigt att tänka på att man även måste modifiera sin SNMP-agent på motsvarande sätt.

### *Enkelhet*

Som vi sa tidigare är en av de stora fördelarna med SNMP dess enkelhet, samma enhetliga hantering etc. Stora nackdelen med SNMP är säkerheten, nyare versioner av SNMP har det gjorts en del åt säkerheten, men i den mest spridda versionen, SNMP version 1, är säkerheten väldigt dålig.

Vid konfiguration av styrprogram och agenter måste man ange gemensamt område, community. Standardvärde är Public, tyvärr använder sig de flesta nätverksadministratörer av detta namn. Det gör i sin tur att jag kan koppla in ett styrprogram som tillhör ”Public”, ta ner valfri SNMP-enhet. Därför bör du som första åtgärd ändra communityname till något annat än Public.

## *Versioner*

Finns tre versioner SNMP version 1, 2 och 3. Version 1 är den mest spridda, kom som sagt 1989. Största problemet med version 1 är som vi har sagt tidigare säkerheten. I version 1 kan man inte med säkerhet identifiera avsändaren av SNMP-meddelanden. Det gör SNMP öppet för attacker som avstängning av enheter i nätverket eller omkonfigurering. Som en följd av detta har många tillverkare av utrustning valt att inte implementera SET-kommandot. Kommandot används nämligen för att påverka konfigurationen på enhet. Version 1 fungerar bara på ett TCP/IP-nätverk.

SNMPv2 som skulle bli uppföljaren accepterades aldrig som en standard. Bidragen orsak var att utvecklarna, Case, McCloghrie, Rose och Waldbusser, inte kunde komma överens om hur säkerheten skulle implementeras. Men SNMPv2 är dock en viktig länk i utvecklingen till nästa version, bla gjorde denna versionen det möjligt till trafik mellan två NMS:er. SNMPv2 innebar även en del förbättringar vad det gäller säkerheten. Bl a så har SNMP-meddelandet modifierats med en sk wrapper. Wrapper:n ser till att meddelandet blir krypterat, säker identifiering av avsändaren men även att kontrollera så att meddelandet kommer fram i tid för att säkerställa så att meddelandet inte har blivit avsiktligt fördröjt eller omsänt. Förbättringar vad det gäller prestanda har också gjorts. Version 1 använder sig av upprepade ReadNext-kommando för att läsa in hela MIB-trädet. I version 2 har ett nytt kommando GetBulk som gör samma sak i en enda förfrågan och ett svar. Version 2 kan andra protokoll än TCP/IP användas.

Utveckligen av SNMPv3 startades 1997, första förslaget till standard kom med RFC 2261 producerat 1998.

## *Hur fungerar det?(Detalj)*

*SNMP kommando*

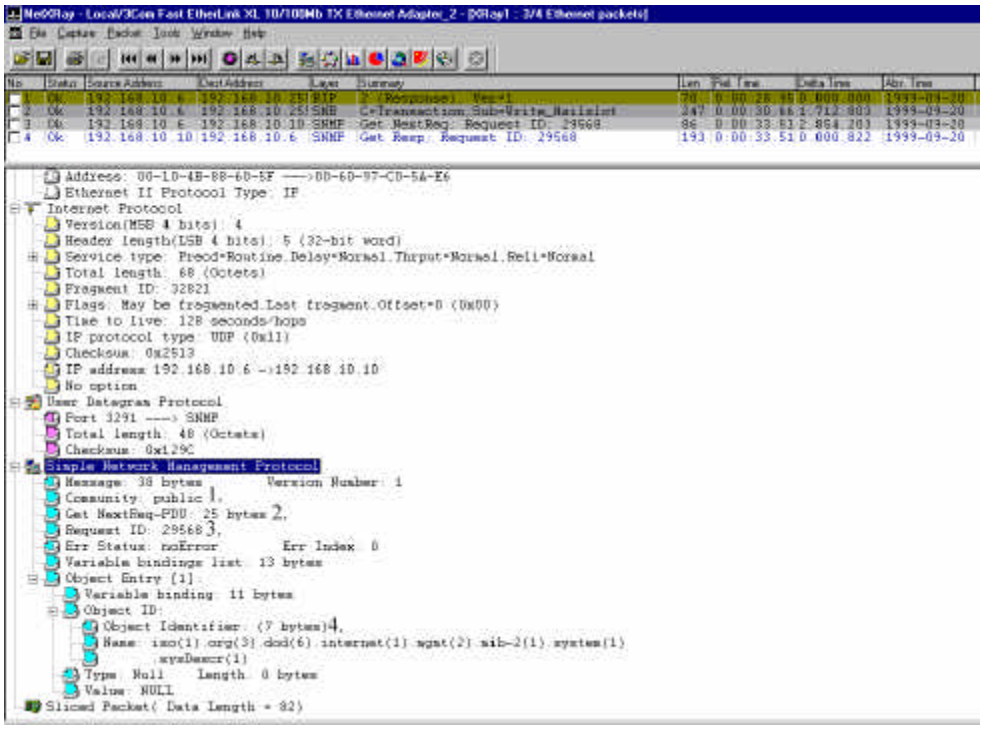
Förutom för NMS-till-NMS kommunikation kan allt SNMP-arbete utföras med fem typer av kommando. Varje kommando finns inuti något som kallas för PDU (Protocol Data Unit), dessa PDU är SNMP-meddelande inkapslat i ett IP-paket.

De fem typer av kommando är:
- GetRequest (skickat av NMS)
- GetNextRequest (skickat av NMS)
- GetResponse (skickat av agent)
- SetRequest (skickat av NMS)
- Trap (skickat av agent)

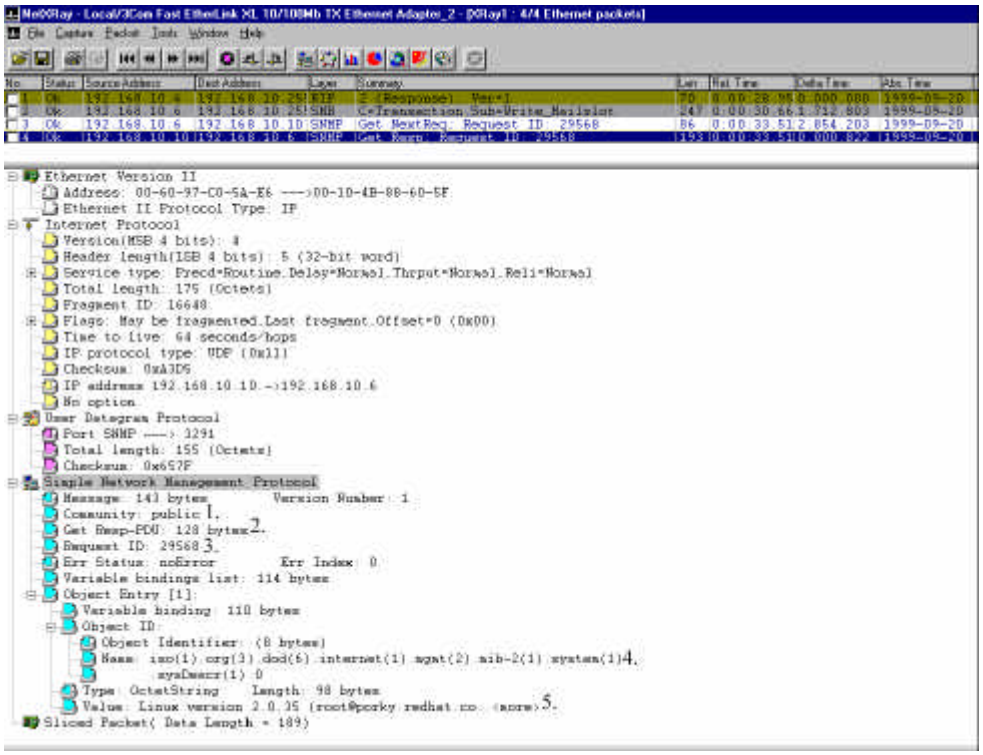
När NMS skickar ut GetRequest eller GetNextRequest då försöker den läsa av variabel i MIB. Denna variabel finns i två typer: ”scalar” eller ”tabular”. Scalar är i form av en lista, tabular är mer komplex representation av agentens information.

*Hämta information från agent*

När NMS vill hämta information från agent sänds ett GetRequest-kommando ut. För att hämta nästa värde från trädet skickas ett GetNextRequest-kommando.




Bilden ovan beskriver vad som händer när jag från NMS-applikation (HP OpenView) begär information från objektidentifieraren 1.3.6.1.2.1.1 . Lägg märke till 1 = community, 2 = kommandot och 3 = id-nummer och 4 = objektidentifierare.



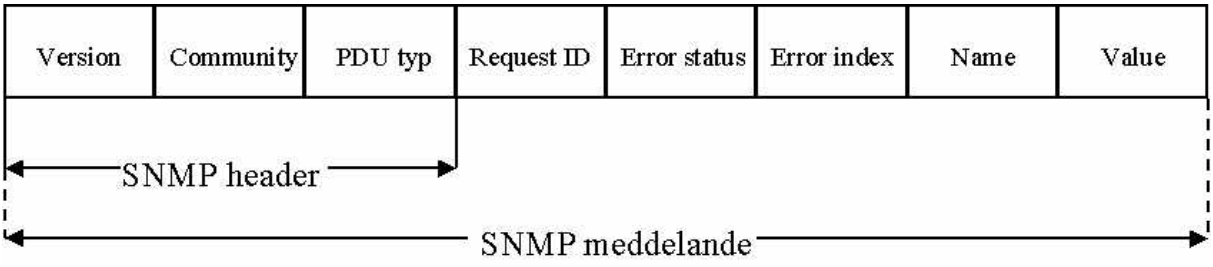

*Agent svarar*

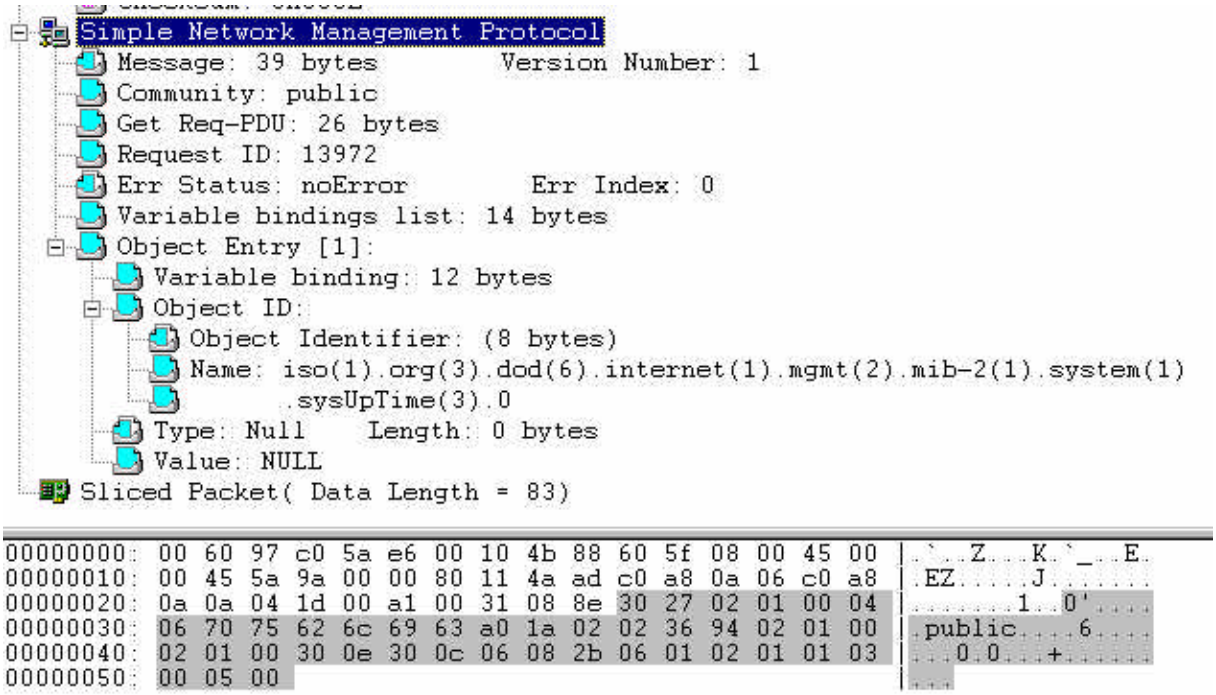
När en agent tar emot GetRequest eller GetNextRequest PDU, inspekterar den värdet av ”MIB-variables”. När agent har hämtat information från MIB, svarar agenten på begäran från

NMS genom att skicka GetResponse PDU. Denna respons innehåller förfrågan plus den efterfrågade informationen. Meddelandet kommer att ha samma identifikationsnummer som förfrågan. Lägg märke till 1 = community, 2 = kommandot och 3 = id-nummer 4 = objektidentifierare och 5 = värdet i MIB.

## *SNMP arkitektur*

SNMP meddelandet består av versionsnummer, ”community name” och ett fält för PDU-typ. Dessa tre fält bildar headern för SNMP-meddelandet. Resten av meddelandet innehåller id-nummer för förfrågan, fel index och olika värden.




*Version*

Version specificerar vilken version som används, det kan vara 0 för SNMPv1, 1 för SNMPv2 eller 2 för SNMPv3.

*Community Name*

Community Name används för att åtskilja meddelanden som tillhör olika administrativa enheter inom nätverket.

*PDU type*

PDU type anger typ av operationer. Det finns fem olika typer av operationer, ”Get” och ”GetNext” används för att efterfråga information från noder, ”Set” modifierar en variabelvärde hos en nod, ”Trap” signalerar händelser såsom att en nod stängs av eller startas upp. Den femte ”GetResponse” används vid svar på de tre första typerna.

*Request ID*

Request ID parar ihop svar med rätt begäran.

*Error Status och Error Index*

Error Status indikerar eventuella fel och feltyper, Error Index specificerar, om fel har uppstått, vilken variabel som felet avser.

## *Portar som används*

Vanligtvis lyssnar NMS på signaler (traps) på UDP port 162. Agenter lyssnar efter NMS-meddelande på UDP port 161. NMS skickar således meddelande på port 161.

När en agent eller NMS vill skicka ett meddelande, kommer den att dynamiskt skapa en socket. Socket består av en länkning mellan UDP-port och en IP-adress. I själva skapade processen kan UDP tilldela vilken UDP-port som helst så länge den finns iblan ”the well-known port numbers”, dvs mellan 1 till 1023.