ĐẠI HỌC THÁI NGUYÊN
KHOA CÔNG NGHỆ THÔNG TIN

TRẦN DUY MINH

GIẢI PHÁP AN NINH TRONG KIẾN TRÚC QUẢN TRỊ MẠNG SNMP

Chuyên ngành: Khoa học máy tính

Mã số: 60.48.01

LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN

Người hướng dẫn: PGS.TS Nguyễn Văn Tam

Thái Nguyên, tháng 12/2008

MỤC LỤC

CÁC THUẬT NGỮ VIẾT TẮT

THUẬT NGỮ, VIẾT TẮT	MÔ TẢ Ý NGHĨA
ARP	Address Ressulation Protocol
ASN.1	Abstract Syntax Notation 1
BER	Basic Encoding Rules
Buffer	Bộ đệm
CA	Certificate Authentication
CHAP	Challenge Handshake Authentication Protocol
Datagram	Đơn vị dữ liệu
DES	Data Encryption Standard
full-duplex	Cơ chế truyền song công
ICMP	Internet Control Message Protocol
IETF	Internet Engineering Task Force
IGMP	Internet Group Message Protocol
ISN	Initial Sequence Number
JNDI	Java Naming Directory Interface
LDAP	Lightweight Directory Access Protocol
MIB	Management Information Base
MSS	Maximum Segment Size
NAS	Network Access Service
NMS	Network Management System
OID	Object identifier
Packet filtering	Bộ lọc gói tin
PAP	Password Authentication Protocol
PDU	Protocol Data Unit
RADIUS	Remote Authentication Dial-In User Service
RARP	Reverse Address Ressulation Protocol
RAS	Remote Access Service
RFC	Requests for Comments
RMON	Remote Network Monitoring
Segment	Đoạn dữ liệu
SGMP	Simple Gateway Management Protocol
SMI	Structure of Management Information
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
TACACS	Terminal Access Controller Access-Control System
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
UDP	User Datagram Protocol

DANH MỤC CÁC HÌNH

ĐẶT VẤN ĐỀ

Công nghệ mạng Internet/Intranet đang phát triển mạnh mẽ và xu hướng tích hợp các mạng không đồng nhất để chia sẻ thông tin cũng xuất hiện ngày càng nhiều. Việc bảo đảm hệ thống mạng phức tạp, có quy mô lớn hoạt động tin cậy, hiệu năng cao, thông tin tin cậy đòi hỏi phải phải có hệ quản trị mạng để thu thập và phân tích một số lượng lớn dữ liệu một cách hiệu quả. Tuy nhiên, thông tin quản trị mạng lại phải truyền trên môi trường Internet, có thể bị thất thoát, thay đổi hay giả mạo cần phải được bảo vệ. Các phiên bản SNMPv1 và SNMPv2 mới chỉ đưa ra giải pháp xác thực yếu dựa trên cộng đồng (community). Chính vì vậy, việc nghiên cứu các giải pháp bảo đảm tính xác thực, tính toàn vẹn, tính mật của các thông điệp quản trị mạng là hết sức cần thiết. Phiên bản SNMPv3 đã ra đời nhằm đáp ứng một phần yêu cầu cấp bách này. Tuy nhiên, việc lựa chọn mô hình thực thi vẫn còn nhiều vấn đề cần giải quyết. Tôi chọn hướng nghiên cứu này mong muốn đóng góp, xây dựng thử nghiệm vào một mô hình cụ thể và qua đó đánh giá khả năng triển khai trong thực tế hệ thống quản trị mạng có độ an ninh cao.

Khuôn khổ luận văn bao gồm 3 chương:

Chương 1: Tổng quan về quản trị và an ninh thông tin trên Internet.

Chương 2: Nghiên cứu giải pháp an ninh mạng SNMP.

Chương 3: *Xây dựng mô hình thử nghiệm.*

Em xin chân thành cảm ơn sự nhiệt tình giúp đỡ của thầy giáo PGS.TS Nguyễn Văn Tam đã giúp em hoàn thành luận văn.

Người thực hiện

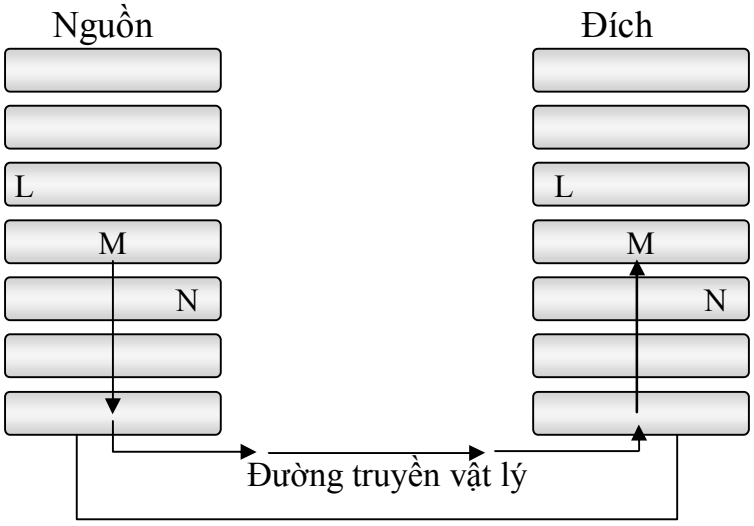
Trần Duy Minh

Chương 1: TỔNG QUAN VỀ QUẢN TRỊ VÀ AN NINH THÔNG TIN TRÊN INTERNET

1.1. Giao thức và dịch vụ Internet

Bộ giao thức là tập hợp các giao thức cho phép sự truyền thông mạng từ một host thông qua mạng đến host khác. Giao thức là một mô tả hình thức của một tập luật và tiêu chuẩn khống chế một khía cạnh đặc biệt trong hoạt động thông tin của các thiết bị trên mạng. Giao thức xác định dạng thức, định thời, tuần tự và kiểm soát lỗi trong hoạt động truyền số liệu. Không có giao thức, máy tính không thể tạo ra hay tái tạo luồng bít đến từ máy tính khác sang dạng ban đầu. Các giao thức điều khiển tất cả các khía cạnh của hoạt động truyền số liệu, bao gồm:

- Mạng vật lý được xây dựng như thế nào.
- Các máy tính được kết nối đến mạng như thế nào.
- Số liệu được định dạng như thế nào để truyền.
- Số liệu được truyền như thế nào.
- Đối phó với lỗi như thế nào.



L, M, N	Các lớp trong mô hình truyền thông
Msource, Mdestination	Các lớp ngang hàng
⟶	Truyền thông ngang hàng
M layer Protocol	Các nguyên tắc thông tin giữa Msource và Mdestination

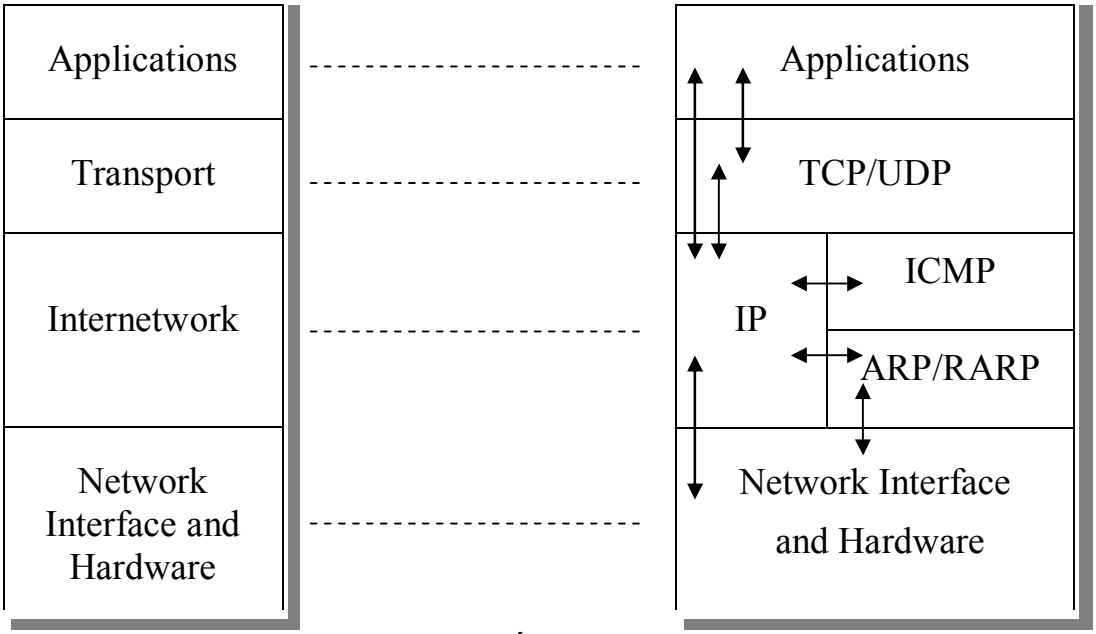
Hình 1.1: Giao thức truyền thông trên máy tính

Các luật mạng này được tạo ra và duy trì bởi nhiều tổ chức và hiệp hội khác nhau. Bao gồm trong các nhóm này là IEEE, ANSI, TIA/EIA và ITU-T (trước đây là CCITT).

1.1.1. Giới thiệu giao thức TCP/IP

Giao thức TCP/IP (Transmission Control Protocol/Internet Protocol) là bộ giao thức cho phép kết nối các hệ thống mạng không đồng nhất với nhau. Ngày nay TCP/IP được sử dụng rộng rãi trong các mạng cục bộ cũng như trên Internet toàn cầu. TCP/IP được xem là giản lược của mô hình tham chiếu OSI với 4 tầng như sau:

+ Tầng liên kết mạng (Network Access Layer)
+ Tầng Internet (Internet Layer)
+ Tầng giao vận (Host-To-Host Transport Layer)
+ Tầng ứng dụng (Application Layer)

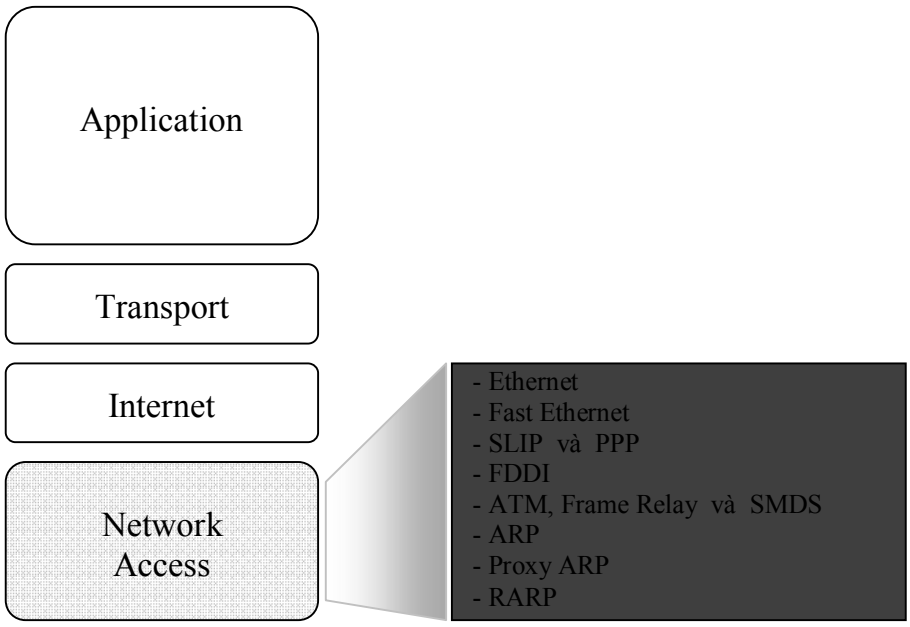


Hình 1.2. Kiến trúc TCP/IP

➢*Tầng liên kết:* Tầng liên kết (còn được gọi là tầng liên kết dữ liệu hay là tầng giao tiếp mạng) là tầng thấp nhất trong mô hình TCP/IP, bao gồm các thiết bị giao tiếp mạng và chương trình cung cấp các thông tin cần thiết để có thể hoạt động, truy nhập đường truyền vật lý qua thiết bị giao tiếp mạng

đó. Nó bao gồm các chi tiết của công nghệ LAN, WAN và tất cả các chi tiết chứa trong lớp vật lý và lớp liên kết số liệu của mô hình OSI.

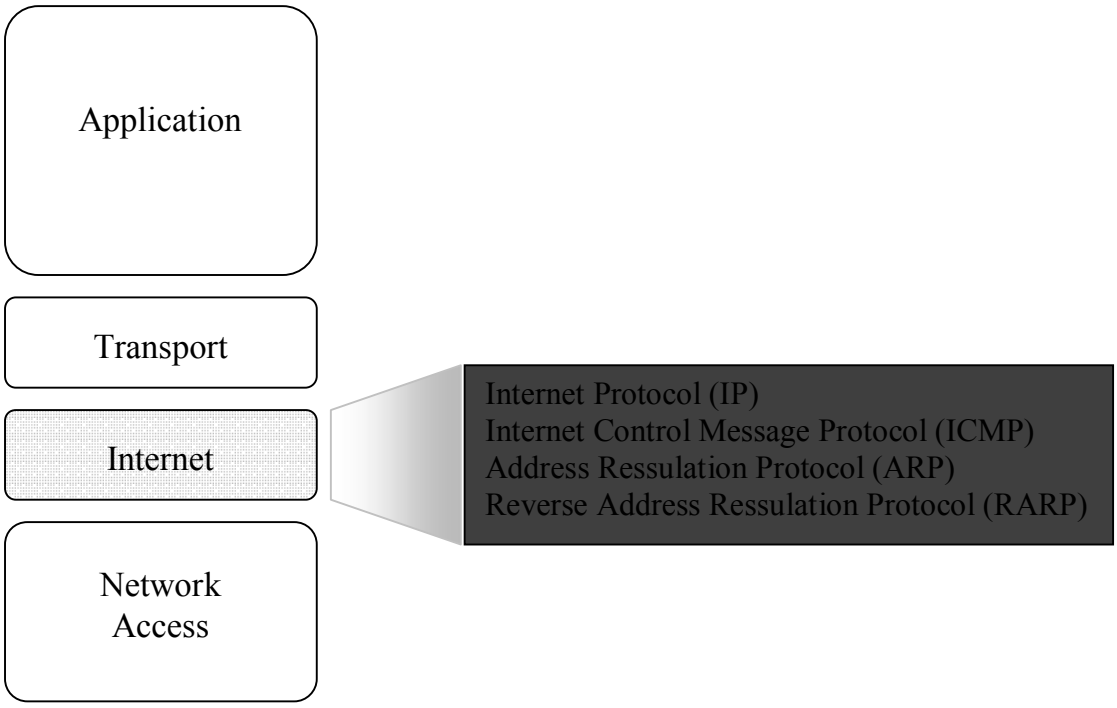
Lớp liên kết định ra các thủ tục để giao tiếp với phần cứng mạng và truy nhập môi trường truyền. Các tiêu chuẩn giao thức modem như SLIP (Serial Line Internet Protocol) và PPP (Point-To-Point Protocol) cung cấp truy xuất mạng thông qua kết nối dùng modem.



Hình 1.3: Các giao thức thuộc lớp Network Access

Chức năng của lớp truy nhập mạng bao gồm ánh xạ địa chỉ IP sang địa chỉ vật lý và đóng gói (encapsulation) các gói IP thành các frame. Căn cứ vào dạng phần cứng và giao tiếp mạng, lớp truy nhập mạng sẽ xác lập kết nối với đường truyền vật lý của mạng.

➢*Tầng Internet*: Tầng Internet (còn gọi là tầng mạng) xử lý qua trình truyền gói tin trên mạng. Các giao thức của tầng này bao gồm: IP (Internet Protocol), ICMP (Internet Control Message Protocol), IGMP (Internet Group Message Protocol). Mục đích của lớp Internet là chọn lấy một đường dẫn tốt nhất xuyên qua mạng cho các gói di chuyển tới đích. Giao thức chính hoạt động tại lớp này là Internet Protocol. Sự xác định đường dẫn tốt nhất và mạch chuyển gói diễn ra tại lớp này.



Hình 1.4: Các giao thức tại lớp Internet

- IP cung cấp conectionless, định tuyến chuyển phát gói theo best-effort. IP không quan tâm đến nội dung của các gói nhưng tìm kiếm đường dẫn cho gói tới đích.

- ICMP (Internet Control Message Protocol): đem đến khả năng điều khiển và chuyển thông điệp.

- ARP (Address Ressulation Protocol): xác định địa chỉ lớp liên kết số liệu (MAC address) khi biết trước địa chỉ IP.

- RARP (Reverse Address Ressulation Protocol): xác định các địa chỉ IP khi biết trước địa chỉ MAC.

IP thực hiện các hoạt động sau:

+ Định nghĩa một gói là một lược đồ đánh địa chỉ.

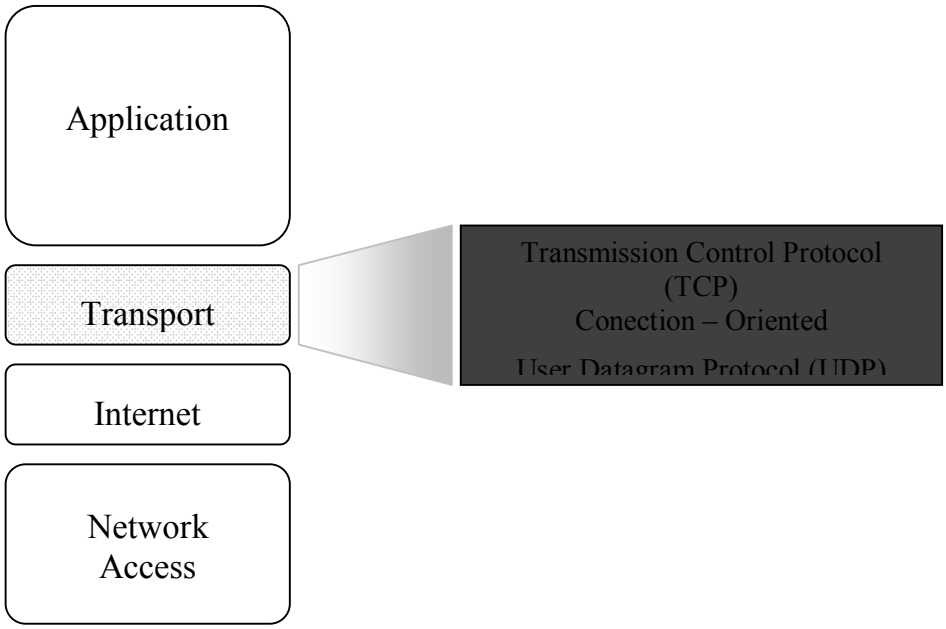
+ Trung chuyển số liệu giữa lớp Internet và lớp truy nhập mạng.

+ Định tuyến chuyển các gói đến host ở xa.

➢*Tầng giao vận*: Tầng giao vận phụ trách luồng giữ liệu giữa hai trạm thực hiện các ứng dụng của tầng trên. Tầng này có hai giao thức chính: TCP (Transmission Protocol), UDP (User Datagram Protocol).

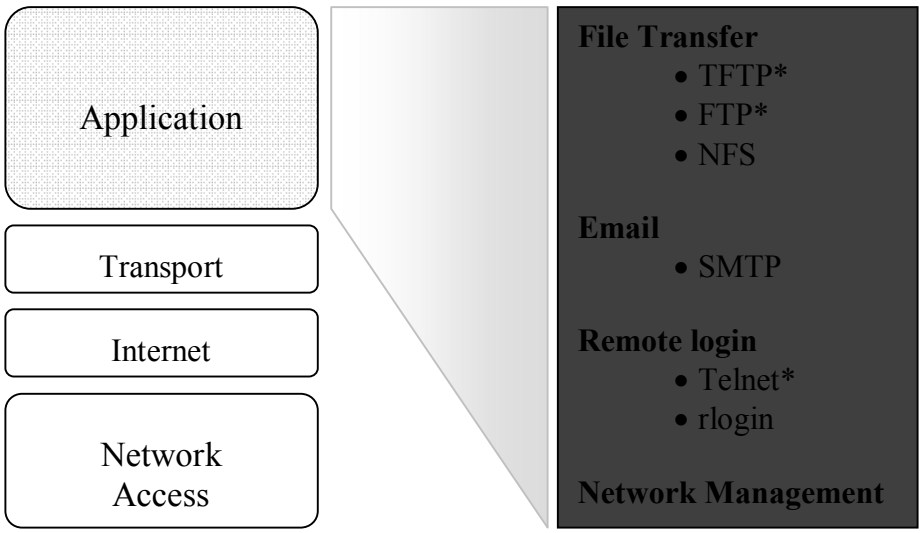
TCP cung cấp luồng dữ liệu tin cậy giữa hai trạm, nó sử dụng các cơ chế như chia nhỏ các gói tin của tầng trên thành các gói tin có kích thước thích hợp cho tầng mạng bên dưới, báo nhận gói tin, đặt hạn chế thời gian time-out để đảm bảo bên nhận biết được các gói tin đã chuyển đi. Do tầng này đảm bảo tính tin cậy, tầng trên sẽ không cần quan tâm đến nữa.

UDP cung cấp một dịch vụ đơn giản hơn cho tầng ứng dụng, nó chỉ gửi các gói tin dữ liệu từ trạm này tới trạm kia mà không đảm bảo các gói tin đến được tới đích. Các cơ chế đảm bảo độ tin cậy cần được thực hiện bởi tầng trên.



Hình 1.5: Các giao thức thuộc lớp Transport

➢*Tầng ứng dụng*: Tầng ứng dụng là tầng trên cùng của mô hình TCP/IP bao gồm các tiến trình và các ứng dụng cung cấp cho người sử dụng để truy cập mạng. Lớp ứng dụng của mô hình TCP/IP kiểm soát các giao thức lớp cao, các chủ đề về trình bày, biểu diễn thông tin, mã hóa và điều khiển hội thoại. Có rất nhiều ứng dụng được cung cấp trong tầng này, mà phổ biến là: Telnet được sử dụng trong mạng truy cập từ xa, FTP (File Transfer Protocol) là dịch vụ truyền tệp, Email – dịch vụ thư tín điện tử, WWW (World Wide Web).



Hình 1.6: Các giao thức thuộc lớp Application

Ý nghĩa của một số dịch vụ:

+ File Transfer Protocol (FTP): là một dịch vụ có tạo cầu nối (conection - oriented) tin cậy, nó sử dụng TCP để truyền các tệp tin giữa các hệ thống có hỗ trợ FTP. Nó hỗ trợ truyền file nhị phân hai chiều và tải các file ASCII.

+ Trivial File Transfer Protocol (TFTP): là một dịch vụ không tạo cầu nối (conectionless) dùng giao thức UDP. TFTP được dùng trên router để truyền các file cấu hình và các Cisco IOS image và để truyền file giữa các hệ thống hỗ trợ TFTP. Nó hữu dụng trong một vài LAN bởi nó hoạt động nhanh hơn FTP trong một môi trường ổn định.

+ Network File System (NFS): là một bộ giao thức hệ thống file phân tán được phát triển bởi Sun Microsystem cho phép truy xuất file đến các thiết bị lưu trữ ở xa như một đĩa cứng qua mạng.

+ Simple Mail Transfer Protocol (SMTP): quản lý các hoạt động truyền e-mail qua mạng máy tính.

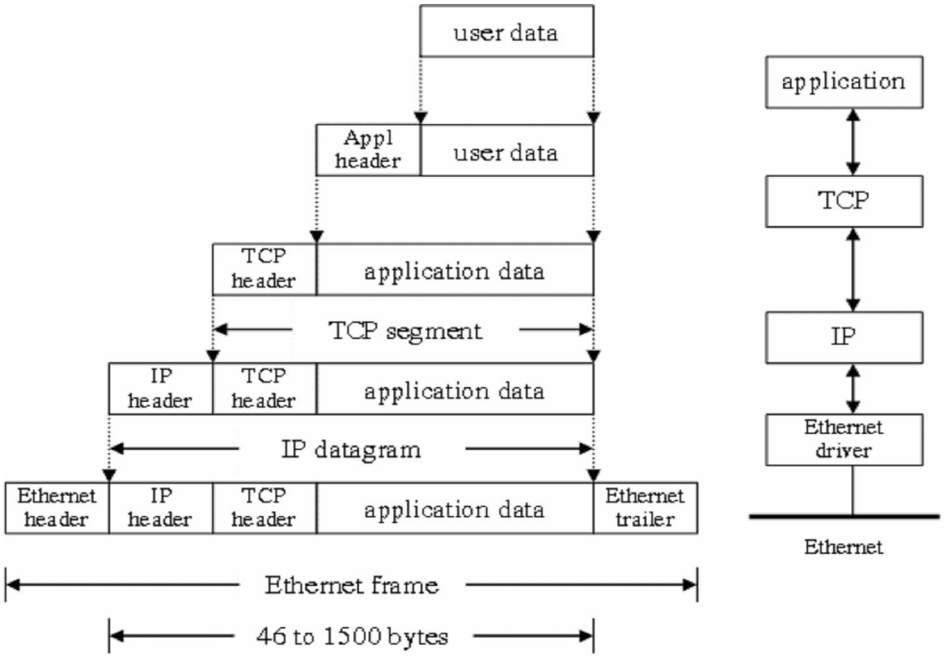
+ Terminal emulation (Telnet): cung cấp khả năng truy nhập từ xa vào các máy tính, thiết bị khác.

+ Simple Network Management Protocol (SNMP): là một giao thức cung cấp phương pháp để giám sát và điều khiển các thiết bị mạng và để quản lý các cấu hình, thu thập thống kê, hiệu suất và bảo mật.

+ Domain Name System (DNS): là một hệ thống được dùng trên Internet để thông dịch tên của các miền (domain) và các node mạng được quảng cáo công khai sang các địa chỉ IP.

**** Quá trình đóng mở gói dữ liệu TCP/IP***

Cũng như mô hình OSI, trong mô hình kiến trúc TCP/IP mỗi tầng có một cấu trúc dữ liệu riêng, độc lập với cấu trúc dữ liệu được dùng ở tầng trên hay tầng dưới kề nó. Khi dữ liệu được truyền từ tầng ứng dụng cho đến tầng vật lý, qua mỗi tầng được thêm phần thông tin điều khiển (Header) đặt trước phần dữ liệu được truyền, đảm bảo cho việc truyền dữ liệu chính xác. Việc thêm thông tin điều khiển vào đầu các gói tin khi đi qua mỗi tầng trong quá trình truyền dữ liệu được gọi là quá trình đóng gói. Quá trình nhận dữ liệu sẽ diễn ra theo chiều ngược lại, khi qua mỗi tầng, các gói tin sẽ tách thông tin điều khiển thuộc nó trước khi chuyển dữ liệu lên tầng trên.



Hình 1.7: Quá trình đóng mở gói dữ liệu TCP/IP

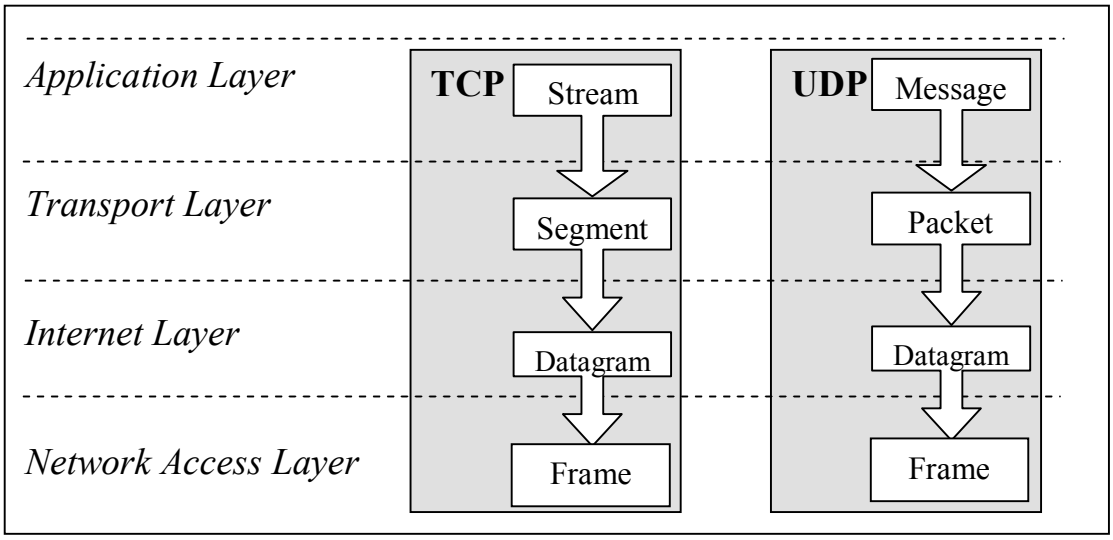
Cũng tương tự như trong mô hình OSI, khi truyền dữ liệu, quá trình tiến hành từ tầng trên xuống tầng dưới, qua mỗi tầng dữ liệu được thêm vào một thông tin điều khiển được gọi là phần header. Khi nhận dữ liệu thì quá trình xảy ra ngược lại, dữ liệu được truyền từ tầng dưới lên và qua mỗi tầng thì phần header tương ứng được lấy đi và khi đến tầng trên cùng thì dữ liệu không còn phần header nữa. Hình 1.8 cho ta thấy lược đồ dữ liệu qua các tầng. Trong hình 1.8 ta thấy tại các tầng khác nhau dữ liệu được mang những thuật ngữ khác nhau:

− Trong tầng ứng dụng dữ liệu là các luồng được gọi là stream.

− Trong tầng giao vận, đơn vị dữ liệu mà TCP gửi xuống tầng dưới gọi là TCP segment.

− Trong tầng mạng, dữ liệu mà IP gửi tới tầng dưới được gọi là IP datagram.

− Trong tầng liên kết, dữ liệu được truyền đi gọi là frame.

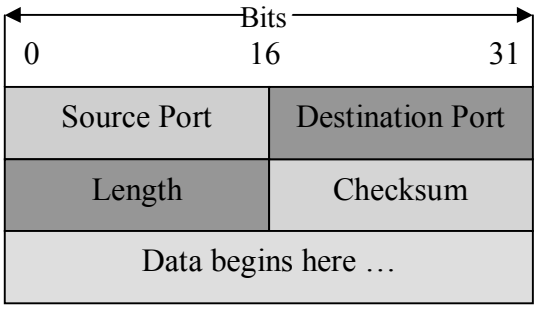


Hình 1.8: Cấu trúc dữ liệu trong TCP/IP

1.1.2. Giao thức UDP

UDP là giao thức không liên kết trong chồng giao thức TCP/IP, cung cấp dịch vụ giao vận không tin cậy, sử dụng thay thế cho TCP trong tầng giao vận. Khác với TCP, UDP không có chức năng thiết lập và giải phóng liên kết, không có cơ chế báo nhận (ACK), không sắp xếp tuần tự các đơn vị

dữ liệu (datagram) đến và có thể dẫn đến tình trạng mất hoặc trùng dữ liệu mà không hề có thông báo lỗi cho người gửi. Khuôn dạng đơn vị dữ liệu của UDP được mô tả như sau:



Hình 1.9: Khuôn dạng UDP datagram

− Số hiệu cổng nguồn (Source Port - 16 bit): số hiệu cổng nơi đã gửi dữ liệu.

− Số hiệu cổng đích (Destination Port - 16 bit): số hiệu cổng nơi dữ liệu được chuyển tới

− Độ dài UDP (Length - 16 bit): độ dài tổng cổng kể cả phần header của gói dữ liệu UDP.

− UDP Checksum (16 bit): dùng để kiểm soát lỗi, nếu phát hiện lỗi thì đơn vị dữ liệu UDP sẽ bị loại bỏ mà không có một thông báo nào trả lại cho trạm gửi.

Các giao thức dùng UDP gồm:

- TFTP (Trivial File Transfer Protocol)
- SNMP (Simple Network Management Protocol)
- DHCP (Dynamic Host Control Protocol)
- DNS (Domain Name System)

UDP có chế độ gán và quản lý các số hiệu cổng (port number) để định danh duy nhất cho các ứng dụng chạy trên một trạm của mạng. Do có ít chức năng phức tạp nên UDP có xu thế hoạt động nhanh hơn so với TCP. Nó thường dùng cho các ứng dụng không đòi hỏi độ tin cậy cao trong giao vận.

1.1.3. Giao thức TCP

TCP và UDP là 2 giao thức ở tầng giao vận và cùng sử dụng giao thức IP trong tầng mạng. Nhưng không giống như UDP, TCP cung cấp một hoạt động truyền dữ liệu song công hoàn toàn (full-duplex) tin cậy và có liên kết. Có liên kết ở đây có nghĩa là 2 ứng dụng sử dụng TCP phải thiết lập liên kết với nhau trước khi trao đổi dữ liệu. Sự tin cậy trong dịch vụ được cung cấp bởi TCP được thể hiện như sau:

− Dữ liệu từ tầng ứng dụng gửi đến được TCP chia thành các đoạn (segment) có kích thước phù hợp nhất để truyền đi.

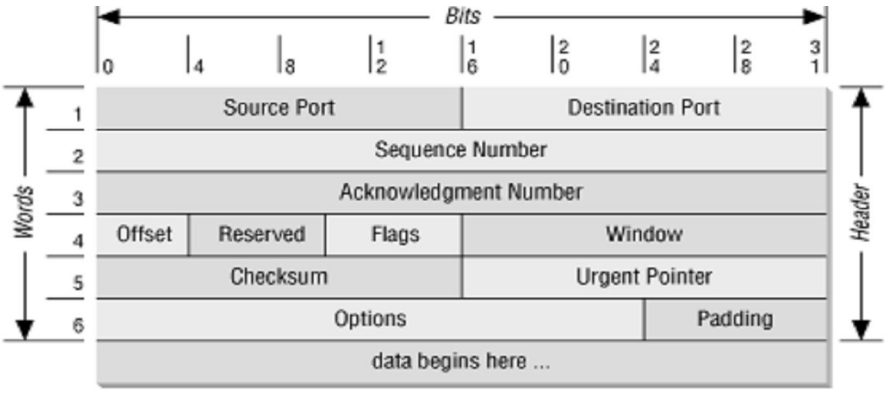
− Khi TCP gửi 1 đoạn, nó duy trì một thời lượng để chờ phúc đáp từ trạm nhận. Nếu trong khoảng thời gian đó phúc đáp không tới được trạm gửi thì đoạn đó được truyền lại.

− Khi TCP trên trạm nhận nhận dữ liệu từ trạm gửi nó sẽ gửi tới trạm gửi 1 phúc đáp tuy nhiên phúc đáp không được gửi lại ngay lập tức mà thường trễ một khoảng thời gian.

− TCP duy trì giá trị tổng kiểm tra (checksum) trong phần Header của dữ liệu để nhận ra bất kỳ sự thay đổi nào trong quá trình truyền dẫn. Nếu 1 đoạn bị lỗi thì TCP ở phía trạm nhận sẽ loại bỏ và không phúc đáp lại để trạm gửi truyền lại đoạn bị lỗi đó.

Giống như đơn vị dữ liệu của IP, các đoạn của TCP có thể tới đích một cách không tuần tự. Do vậy TCP ở trạm nhận sẽ sắp xếp lại dữ liệu và sau đó gửi lên tầng ứng dụng đảm bảo tính đúng đắn của dữ liệu.

Khi dữ liệu IP bị trùng lặp TCP tại trạm nhận sẽ loại bỏ dữ liệu trùng lặp đó.



Hình 1.10: Khuôn dạng TCP segment

TCP cũng cung cấp khả năng điều khiển luồng. Mỗi đầu của liên kết TCP có vùng đệm (buffer) giới hạn do đó TCP tại trạm nhận chỉ cho phép trạm gửi truyền một lượng dữ liệu nhất định (nhỏ hơn không gian đệm còn lại). Điều này tránh xảy ra trường hợp trạm có tốc độ cao chiếm toàn bộ vùng đệm của trạm có tốc độ chậm hơn.

Khuôn dạng của một đoạn TCP được mô tả trong hình 1.10

Các tham số trong khuôn dạng trên có ý nghĩa như sau:

− Source Port (16 bits) là số hiệu cổng của trạm nguồn .

− Destination Port (16 bits) là số hiệu cổng trạm đích .

− Sequence Number (32 bits) là số hiệu byte đầu tiên của đoạn trừ khi bit SYN được thiết lập. Nếu bit SYN được thiết lập thì sequence number là số hiệu tuần tự khởi đầu ISN (Initial Sequence Number) và byte dữ liệu đầu tiên là ISN + 1. Thông qua trường này TCP thực hiện việc quản lí từng byte truyền đi trên một kết nối TCP.

− Acknowledgment Number (32 bits). Số hiệu của đoạn tiếp theo mà trạm nguồn đang chờ để nhận và ngầm định báo nhận tốt các segment mà trạm đích đã gửi cho trạm nguồn .

− Header Length (4 bits). Số lượng từ (32 bits) trong TCP header, chỉ ra vị trí bắt đầu của vùng dữ liệu vì trường Option (tùy chọn) có độ dài thay đổi. Header length có giá trị từ 20 đến 60 byte .

− Reserved (6 bits). Dành để dùng trong tương lai .

− Control bits : các bit điều khiển

URG : xác định vùng con trỏ khẩn có hiệu lực.

ACK : vùng báo nhận ACK Number có hiệu lực.

PSH : chức năng PUSH.

RST : khởi động lại liên kết.

SYN : đồng bộ hoá các số hiệu tuần tự (Sequence number).

FIN : không còn dữ liệu từ trạm nguồn.

− Window size (16 bits): cấp phát thẻ để kiểm soát luồng dữ liệu (cơ chế cửa sổ trượt). Đây chính là số lượng các byte dữ liệu bắt đầu từ byte được chỉ ra trong vùng ACK number mà trạm nguồn sẵn sàng nhận.

− Checksum (16 bits). Mã kiểm soát lỗi cho toàn bộ segment cả phần header và dữ liệu.

− Urgent Pointer (16 bits). Con trỏ trỏ tới số hiệu tuần tự của byte cuối cùng trong dòng dữ liệu khẩn cho phép bên nhận biết được độ dài của dữ liệu khẩn. Vùng này chỉ có hiệu lực khi bit URG được thiết lập.

− Option (độ dài thay đổi). Khai báo các tuỳ chọn của TCP trong đó thông thường là kích thước cực đại của 1 segment: MSS (Maximum Segment Size).

− TCP data (độ dài thay đổi). Chứa dữ liệu của tầng ứng dụng có độ dài ngầm định là 536 byte. Giá trị này có thể điều chỉnh được bằng cách khai báo trong vùng tùy chọn.

Các giao thức dùng TCP bao gồm:

- FTP (File Transfer Protocol)
- HTTP (Hypertext Transfer Protocol)
- SMTP (Simple Mail Transfer Protocol)
- Telnet

1.2. Các mô hình quản trị mạng SNMP

Giao thức TCP/IP trên nền Ethernet hết sức thông dụng trên thị trường truyền thông hiện nay. Sự thành công của các công nghệ trên nền Ethernet một phần là do sự hợp tác rất tích cực trong quá trình phát triển các chuẩn chung. Sự thành công này cũng sẽ tạo ra những sức mạnh mới trên những cơ sở hạ tầng sẵn có như hệ thống cáp, kiến trúc mạng, khuôn dạng gói tin và các trình điều khiển vốn đã được cài đặt trong các mạng Ethernet hiện có.

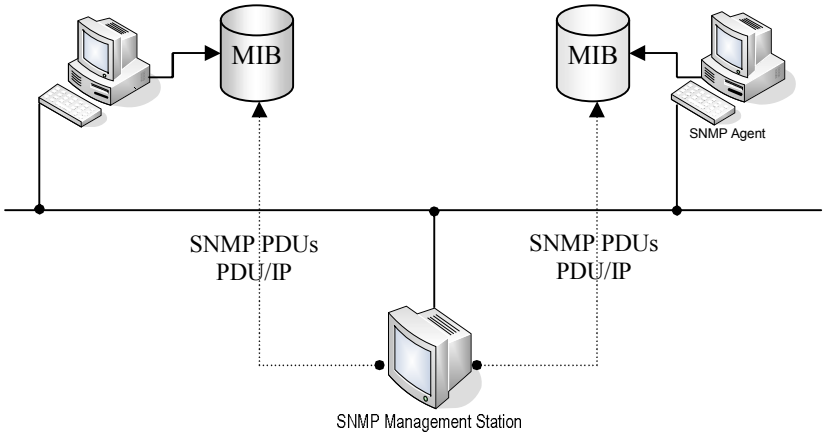
Quản lý mạng là nhiệm vụ đầy thử thách, quy mô mạng càng lớn càng phức tạp. Hiện nay, hầu hết phần tử mạng có các module quản lý riêng nên việc quản lý bị phân tán. Xu hướng tương lai là tập trung hóa hệ thống quản lý mạng bằng việc tích hợp tất cả phần tử mạng trong một cơ sở dữ liệu tập trung và chia sẻ cho nhiều người quản trị mạng.

SNMP là giao thức quản lý mạng hiện được dùng rất phổ biến trên mạng TCP/IP. Sau đây là hai mô hình quản lý mạng sử dụng giao thức SNMP điển hình.

1.2.1. Quản lý mạng Microsoft sử dụng SNMP

Các mô hình quản lý mạng truyền thống chạy trên hệ điều hành của Microsoft đa số sử dụng giao thức SNMP, trong đó chia làm 4 thành phần:

- Nút được quản lý (managed node)
- Trạm quản lý (management station)
- Thông tin quản lý (management information)
- Giao thức quản lý (management protocol)



Hình 1.11: Quản lý mạng Microsoft sử dụng SNMP

- Nút được quản lý có thể là máy tính, bộ định tuyến, bộ chuyển mạch, cầu nối, máy in hoặc các thiết bị mạng khác có khả năng liên lạc với bên ngoài mạng. Mỗi nút chạy phần mềm quản lý gọi là SNMP agent. Mỗi agent duy trì một cơ sở dữ liệu cục bộ các biến mô tả trạng thái, lịch sử và tác vụ ảnh hưởng lên nó.

- Trạm quản lý chứa một hoặc nhiều tiến trình liên lạc với agent trên mạng, phát những câu lệnh và nhận kết quả. Hình 1.11 trình bày mô hình quản lý mạng Microsoft thông qua giao thức SNMP.

Trong hình 1.11, cơ sở dữ liệu MIB (Management Information Base) tập hợp tất cả các đối tượng trong một mạng, nó định ra những biến mà các phần tử mạng cần duy trì.

Trạm quản lý (management station) tương tác với agent qua giao thức SNMP. Giao thức SNMP gồm 5 tác vụ và mỗi tác vụ được mã hóa trong một đơn vị dữ liệu PDU (Protocol Data Unit) riêng biệt và được chuyển qua mạng bằng giao thức UDP. Đó là các tác vụ:

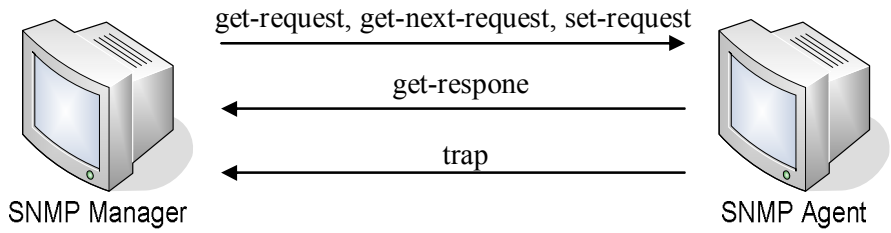
• **Get-request:** lấy giá trị của một hoặc nhiều biến.

• **Get-next-request:** lấy giá trị của biến kế tiếp.

• **Set-request:** đặt giá trị của một hoặc nhiều biến.

• **Get-response:** trả về giá trị của một hoặc nhiều biến sau khi phát lệnh get-request hoặc get-next-request, hoặc set-request.

• **Trap:** gửi cảnh báo cho agent quản lý khi có biến cố xảy ra trên máy agent.



Hình 1.12: Các tác vụ SNMP

Hình 1.12 minh họa 5 tác vụ liên lạc giữa agent quản lý và máy agent, trong đó SNMP sử dụng port 161 cho các lệnh get-request, get-next-request, set-request và get-response, riêng lệnh trap thì sử dụng port 162. Để minh họa cách thức SNMP làm việc như thế nào, chúng ta xem ví dụ ở hình 1.13.

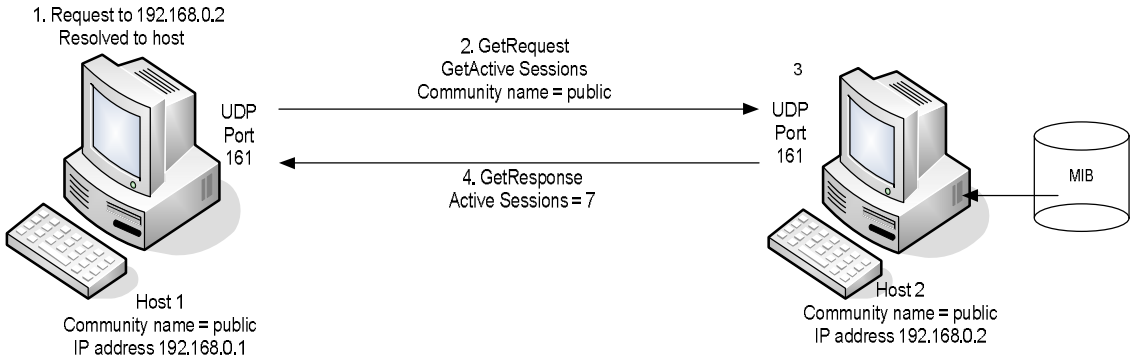
Giả sử có một ứng dụng quản lý SNMP chạy trên máy host 1 yêu cầu số phiên kích hoạt từ một máy Microsoft SNMP agent là host 2.

+ Trình quản lý SNMP sử dụng tên máy (host name) để gửi yêu cầu qua cổng dịch vụ UDP 161. Tên máy sẽ được phân giải bằng cách sử dụng các file HOST, DNS hoặc WINS v.v...

+ Một message SNMP chứa lệnh get-request phát ra để phát hiện số phiên kích hoạt với tên community name là public.

+ Máy host 2 nhận message và kiểm tra tên nhóm làm việc chung (community name). Nếu tên nhóm sai hoặc message bị hỏng thì yêu cầu từ phía máy host 1 bị hủy bỏ. Nếu tên nhóm đúng và message hợp lệ thì kiểm tra địa chỉ IP để đảm bảo nó được quyền truy nhập message từ agent host 1.

+ Sau đó, phiên kích hoạt được tạo (ví dụ là phiên số 7) và trả thông tin về cho agent quản lý SNMP.



Hình 1.13: Cách thức SNMP làm việc

Nhược điểm:

- Vì 4 trong 5 message SNMP là các nghi thức hồi đáp đơn giản (agent gửi yêu cầu, máy agent phản hồi kết quả) nên SNMP sử dụng giao thức UDP. Điều này nghĩa là một yêu cầu từ agent có thể không đến được máy agent và hồi đáp từ máy agent có thể không trả về cho agent. Vì vậy agent cần cài đặt thời gian hết hạn (timeout) và cơ chế phát lại.

- Quản lý mạng dựa trên SNMP có mức bảo mật thấp. Vì dữ liệu không mã hóa và không có thiết lập cụ thể để ngưng bất kỳ truy nhập mạng trái phép nào khi tên community name và địa chỉ IP bị sử dụng để gửi yêu cầu giả mạo tới agent.

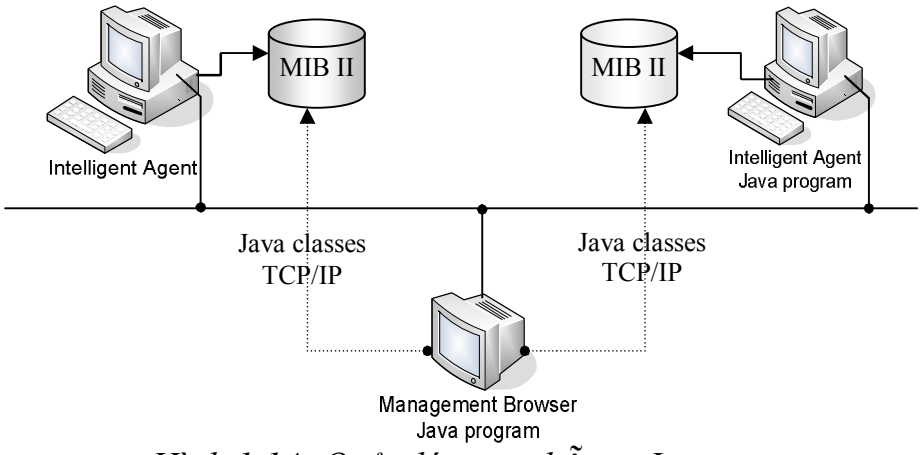
- Quản lý mạng dựa trên SNMP có mức khả chuyển thấp giữa các kiến trúc khác nhau. Vì cấu trúc thông tin quản lý của SNMP chỉ hỗ trợ giới hạn các kiểu dữ liệu.

- Không thân thiện.

1.2.2. Quản lý mạng trên môi trường Java

Sun Microsystem đã hỗ trợ một phương thức quản lý mạng dựa trên môi trường Java. Kiến trúc Java sử dụng giao thức SNMP như giao thức quản lý mạng gồm hai thành phần: trình duyệt quản lý chạy trên hệ thống NMS (Network Management System) và các máy Java thông minh chạy trên các phần tử mạng gọi là các agent thông minh. Dữ liệu liên lạc giữa trình duyệt và thực thể agent được định nghĩa như các lớp đối tượng trong

cơ sở dữ liệu MIB, hoặc được định dạng theo cú pháp ASN.1 (Abstract Syntax Notation 1). Nó được mã hóa để truyền trên mạng dựa trên luật mã hóa cơ sở BER (Basic Encoding Rules). Hình 1.14 minh họa cơ chế quản lý mạng hỗ trợ Java.



Hình 1.14: Quản lý mạng hỗ trợ Java

Ưu điểm:

- Trình duyệt và các agent thực hiện liên lạc với nhau dựa trên những chương trình hoặc lớp Java được mã hóa dưới dạng byte-code và thực thi thông qua các trình thông dịch Java cài sẵn. Vì vậy cơ chế mã hóa theo luật BER của SNMP không cần thiết do tự thân các lớp Java đã mã hóa dưới định dạng byte-code.

- Các đơn vị dữ liệu PDU được thay bởi các lớp Java để chuyển lệnh và dữ liệu.

- Giao thức UDP/IP được thay bởi giao thức TCP/IP.

- Cơ sở dữ liệu theo chuẩn MIB II được hỗ trợ cho các agent.

- Đặc trưng bảo mật vốn có trong mã Java byte-code cung cấp thêm một vỏ bọc an ninh trong quản lý thông tin xuyên mạng.

1.2.3. Cơ chế quản lý mạng tập trung theo mô hình DEN

Một cơ chế mới trong quản lý mạng là ứng dụng mô hình mạng thư mục DEN (Directory Enabled Network) kết hợp giao thức lưu trữ và truy nhập thư mục LDAP (Lightweight Directory Access Protocol) để tập trung

thông tin mạng cần quản lý trong một cơ sở dữ liệu duy nhất nhưng được khai thác sử dụng trên toàn mạng. DEN là một đặc tả lưu trữ thông tin dưới hình thức các lớp trong một cơ sở dữ liệu thư mục tập trung theo giao thức LDAP. Giao thức này hiện đang được nhiều tổ chức, công ty phát triển và hỗ trợ trong các sản phẩm và dịch vụ của mình như trong các thiết bị đầu cuối, hệ điều hành v.v...

Hiện tại, có nhiều cách xây dựng cơ chế quản lý mạng tập trung, trong đó nổi bật là cách sử dụng gói dịch vụ JNDI (Java Naming Directory Interface) được cung cấp sẵn của Sun Microsystem để cài đặt ứng dụng. Thông qua ứng dụng được xây dựng trên nền tảng JNDI người quản trị có thể cập nhật thông tin khi có thêm một phần tử mới trên mạng hoặc tìm kiếm thông tin khi có nhu cầu giám sát, kiểm tra thông tin của một phần tử mạng bất kỳ. Tất cả thông tin này đều được lưu trữ trong một cơ sở dữ liệu thư mục tập trung duy nhất trên mạng và chỉ truy nhập thông qua giao thức LDAP.

Tóm tắt các bước triển khai ứng dụng quản lý mạng dựa trên JNDI như sau:

• Sử dụng đặc tả DEN để mô tả thông tin các phần tử mạng dưới dạng các lớp đối tượng (gồm cả thiết bị mạng, các lớp ứng dụng/dịch vụ mạng và các "hành vi ứng xử" giữa các phần tử mạng).

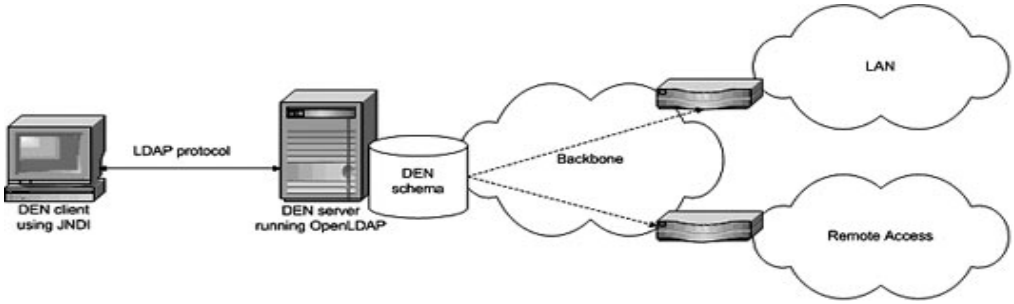
• Thiết lập máy chủ cơ sở dữ liệu thư mục LDAP để lưu các thông tin mạng DEN.

• Sử dụng giao diện lập trình JNDI để cài đặt DEN.

Mô hình quản lý mạng thông qua cơ sở dữ liệu các lớp đối tượng DEN được thể hiện trong hình 1.15.

Ứng dụng DEN cho phép máy khách DEN truy nhập dịch vụ tên và thư mục để liên lạc và tìm các đối tượng và thuộc tính của nó được định nghĩa trong các lớp DEN. JNDI là gói Java tùy chọn cung cấp ngữ cảnh và

giao diện ngữ cảnh thư mục được sử dụng bởi máy khách DEN. JNDI cung cấp truy nhập mức thấp tới giao thức LDAP dùng liên lạc giữa các ứng dụng client và server. Các đối tượng DEN với những thuộc tính và liên kết có thể tích hợp trong một ngữ cảnh thư mục đơn gọi là lược đồ (schema). Thông tin này được lưu trong máy chủ phục vụ có cài đặt phần mềm quản trị LDAP (chẳng hạn OpenLDAP trên nền Linux hoặc Actice Directory trên nền MS Windows).



Hình 1.15: Quản lý mạng qua CSDL các lớp đối tượng DEN

- Dịch vụ tên và thư mục đóng vai trò quan trọng trong mạng qua việc cung cấp đa dạng các thông tin dùng chung về người dùng, máy tính, mạng, dịch vụ và ứng dụng. Các ứng dụng có thể chia sẻ dùng chung không gian lưu trữ cung cấp bởi thư mục. Điều này giúp cho các ứng dụng cài đặt qua mạng dễ dàng và phù hợp hơn.

- DEN định nghĩa một cách thức quản lý mạng hơn là cách quản lý một phần tử mạng (như kiểu quản lý SNMP). Bằng cách tập trung thông tin tại một điểm, DEN giúp cho người quản trị quản lý, bảo dưỡng và kiểm soát mạng một cách dễ dàng.

1.3. Vấn đề bảo đảm an ninh truyền thông trên Internet

1.3.1. Khái niệm về đảm bảo an ninh truyền thông

Mạng Internet đã được phổ cập khắp thế giới do vậy việc bảo vệ tài nguyên thông tin trên mạng là cấp thiết. Vấn đề an ninh mạng càng trở nên cấp thiết để chống các hacker đột nhập vào hệ thống, ăn cắp thông tin và làm tê liệt hệ thống. Mục tiêu của việc đảm bảo an ninh trên mạng là:

+ Tính bảo mật (confidentiality): Bảo đảm dữ liệu không bị sử dụng bởi người không có thẩm quyền.

+ Tính xác thực (Authentication): Kiểm tra tính hợp pháp của người sử dụng.

+ Tính không thể chối cãi (nonrepudiation): Các thực thể tham gia không thể chối bỏ cam kết.

+ Tính toàn vẹn (Integrity): Thông tin không bị sai lệch, sửa đổi.

Quá trình xử lý, phân tích, tổng hợp và bảo mật thông tin là hai mặt của một vấn đề không thể tách rời nhau. Ngay từ khi máy tính ra đời, cùng với nó là sự phát triển ngày càng lớn mạnh và đa dạng của các hệ thống thông tin người ta đã nghĩ ngay đến các giải pháp đảm bảo an toàn cho hệ thống thông tin của mình.

Chúng ta phải kiểm soát các vấn đề an toàn mạng theo các mức khác nhau đó là:

- Mức mạng: Ngăn chặn kẻ xâm nhập bất hợp pháp vào hệ thống mạng.

- Mức Server: Kiểm soát quyền truy nhập, các cơ chế bảo mật, quá trình nhận dạng người dùng, phân quyền truy cập, cho phép các tác vụ.

- Mức cơ sở dữ liệu: Kiểm soát ai? Được quyền như thế nào? với mỗi cơ sở dữ liệu.

- Mức trường thông tin: Trong mỗi cơ sở dữ liệu kiểm soát được mỗi trường dữ liệu chứa thông tin khác nhau có quyền truy cập khác nhau.

- Mức mật mã: Mã hóa toàn bộ file dữ liệu theo một phương pháp nào đó và chỉ cho phép người có "chìa khóa" mới có thể sử dụng được file dữ liệu.

Trong thời gian gần đây, số vụ xâm nhập trái phép vào hệ thống thông tin qua mạng Internet và Intranet ngày càng tăng. Có nhiều nguyên nhân dẫn tới việc các mạng bị tấn công nhiều hơn, trong số những nguyên nhân chính

có thể kể đến xu hướng chuyển sang môi trường tính toán client/server (chủ/khách), các ứng dụng thương mại điện tử, việc hình thành các mạng Intranet của các công ty với việc ứng dụng công nghệ Intranet vào các mạng kiểu này dẫn tới xóa nhòa danh giới giữa phần bên ngoài (Internet) và phần bên trong (Intranet) của mạng. Tạo nên những nguy cơ mới về an toàn thông tin. Cũng cần lưu ý rằng những nguy cơ mất an toàn thông tin không chỉ do tấn công từ bên ngoài mà một phần lớn chính là từ nội bộ: sai sót của người sử dụng, ý thức bảo mật kém,…

Môi trường mạng là khá phức tạp, nhiều người sử dụng và phân tán về mặt địa lý nên an toàn thông tin trên mạng là một công việc vô cùng khó khăn và phức tạp. Nó đòi hỏi phải sử dụng nhiều giải pháp khác nhau từ cơ bản đến phức tạp, tuỳ theo lượng thông tin cần bảo vệ và khả năng cho phép của từng hệ thống cụ thể.

1.3.2. Một số giải pháp

- Kiểm soát đăng ký tên/mật khẩu truy cập mạng.
- Kiểm soát việc truy nhập tài nguyên mạng và quyền hạn trên tài nguyên đó.
- Mã hoá dữ liệu truyền trên mạng (bảo mật thông tin).
- Ngăn cản truy nhập vật lý bất hợp pháp vào hệ thống (bảo vệ vật lý).
- Sử dụng bức tường lửa (filewall) để ngăn cách mạng nội bộ với thế giới bên ngoài hoặc giữa các mạng nội bộ với nhau.

1.3.4. Các thành phần thường gặp trong bức tường lửa

+ Bộ lọc gói tin (Packet filtering): cho phép hay ngăn cấm các gói tin khi chúng truyền từ mạng này sang mạng khác theo địa chỉ IP.

+ Pháo đài bảo vệ (Bastion host): hệ thống máy tính có an ninh cao đặt ở điểm truy cập vào/ra mạng cần được bảo vệ.

+ Máy phục vụ uỷ quyền (Proxy Server): thay mặt người dùng của mạng được bảo vệ và giao tiếp với các máy dịch vụ ở ngoài mạng được bảo vệ.

Hình 1.16:Mô hình các mức bảo vệ an toàn

Chương 2: GIẢI PHÁP AN NINH MẠNG SNMP

2.1. Giao thức quản trị mạng SNMP

SNMP (Simple Network Management Protocol): là giao thức được sử dụng rất phổ biến để giám sát và điều khiển thiết bị mạng như switch, router, bridge... Sử dụng trong các hệ quản trị như Unix, Windows, Printers, Modem racks, power supplies và các thiết bị khác. Với những văn phòng nhỏ chỉ có vài thiết bị mạng và đặt tập trung một nơi thì có lẽ chúng ta không thấy được lợi ích của SNMP. Nhưng với các hệ thống mạng lớn, thiết bị phân tán nhiều nơi và bạn cần phải ngồi một chỗ mà có thể quản lý tất cả thiết bị mới thấy được lợi ích của SNMP. Microsoft Windows Server 2003 cung cấp phần mềm SNMP agent để có thể làm việc với phần mềm quản lý SNMP từ nhà cung cấp thứ 3 nhằm giám sát các trạng thái của thiết bị quản lý và các ứng dụng.

Cốt lõi của SNMP là tập hợp quá trình hoạt động của các thiết bị giám sát làm tăng khả năng quản trị hệ thống. Ví dụ như: có thể sử dụng SNMP để tắt một thiết bị ghép nối nào đó trên router, hay kiểm tra tốc độ của cổng trên router. SNMP có thể sử dụng để cảnh báo khi nhiệt độ của Switch trong hệ thống mạng quá cao.

SNMP là một giao thức thuộc lớp ứng dụng làm phương tiện trao đổi các thông tin quản lý giữa các thiết bị mạng. SNMP cho phép người quản trị mạng quản lý hiệu suất mạng, tìm và giải quyết các vấn đề mạng, cũng như hoạch định cho sự phát triển mạng. SNMP dùng UDP như là một giao thức vận chuyển cho nó.

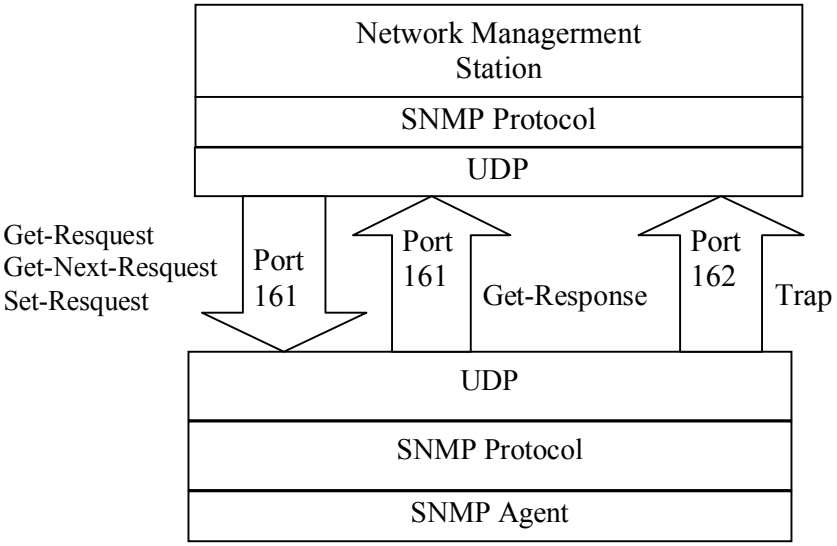
2.1.1. Giới thiệu giao thức SNMP.

Tổ chức IETF (Internet Engineering Task Force) đánh giá cao vai trò của SNMP trong quản trị mạng Internet. IETF đã đưa ra một loạt các RFC (Requests for Comments) mà ở đó các giao thức hầu hết dựa trên cơ sở IP.

Giao thức SNMP được thiết kế để cung cấp một phương thức đơn giản để quản lý tập trung mạng TCP/IP. Nếu bạn muốn quản lý các thiết bị từ 1 vị trí tập trung, giao thức SNMP sẽ vận chuyển dữ liệu từ client (thiết bị mà bạn đang giám sát) đến server nơi mà dữ liệu được lưu trong log file nhằm phân tích dễ dàng hơn. Các phần mềm ứng dụng dựa trên giao thức SNMP như: Tivoli của IBM, MOM của Microsft và HP Openview vv…

Giao thức SNMP là giao thức đã được thị trường chấp nhận trong thời gian rất ngắn. Điều đó là sự chứng minh tốt nhất cho ưu điểm của nó. Giao thức quản lý mạng đơn giản SNMP giúp người quản trị xác định và sửa chữa các vấn đề trong TCP/IP internet. Người quản lý thực thi SNMP client trên máy tính cục bộ của họ, máy tính PC chẳng hạn và sử dụng client để liên lạc với một hoặc nhiều SNMP server nào thực thi trên máy tính ở xa (thường là các gateway). SNMP sử dụng mô hình fetch-store, trong đó mỗi server duy trì một tập hợp các biến khái niệm để chứa các số liệu thống kê đơn giản, như là đếm số packet nhận được, cũng như các biến phức tạp tương ứng với các cấu trúc dữ liệu TCP/IP, như là RARP cache và các bảng định tuyến IP.

Giao thức SNMP nằm ở tầng ứng dụng nó làm dễ dàng việc trao đổi thông tin giữa các thiết bị mạng. Nó hoạt động dựa trên tầng UDP của giao thức IP. Về tập lệnh, giao thức SNMP chỉ có 5 lệnh cơ bản để trao đổi thông tin giữa trạm quản lý và các agent là: Get-Request, Get-Next-Request, Set-Request, Get-Response và Trap. Đây chính là một ưu điểm của SNMP, do cấu trúc đơn giản nên dễ cài đặt.



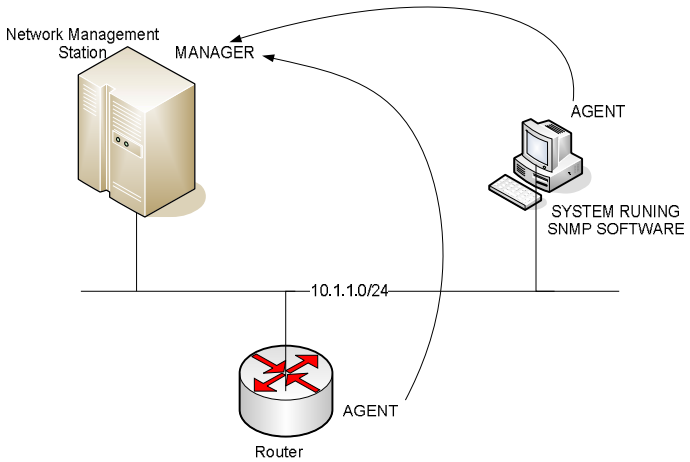
Hình 2.1: Lưu đồ giao thức SNMP

- SNMP sử dụng giao thức UDP

UDP là đối lập với TCP. UDP nhanh hơn, nhưng không tin cậy. Nó thi hành và sử dụng đơn giản hơn là TCP. Tuy nhiên nó cung cấp nhiều chức năng cho phép 1 trạm quản lý tập trung có thể liên lạc với agent từ xa được đặt ở bất kì thiết bị được quản lý nào mà nó có thể liên lạc tới. Ngoài ra việc sử dụng UDP sẽ giảm độ trễ trong mạng so với việc sử dụng TCP.

- Hoạt động của SNMP

Các nhân tố chính trong SNMP: NMS, manager và agent. Manager là các phần mềm quản lý như HP Openview. Agent là các phần mềm SNMP chạy trong 1 hệ thống máy khách mà bạn đang giám sát.



Hình 2.2: Quá trình hoạt động của SNMP

- *Network Management Station (NMS)*

Manager cũng được gọi là NMS. Các chức năng khác của NMS bao gồm các đặc tính báo cáo, network topology mapping và lập tài liệu, các công cụ cho phép bạn giám sát traffic (lưu thông) trên mạng vv…

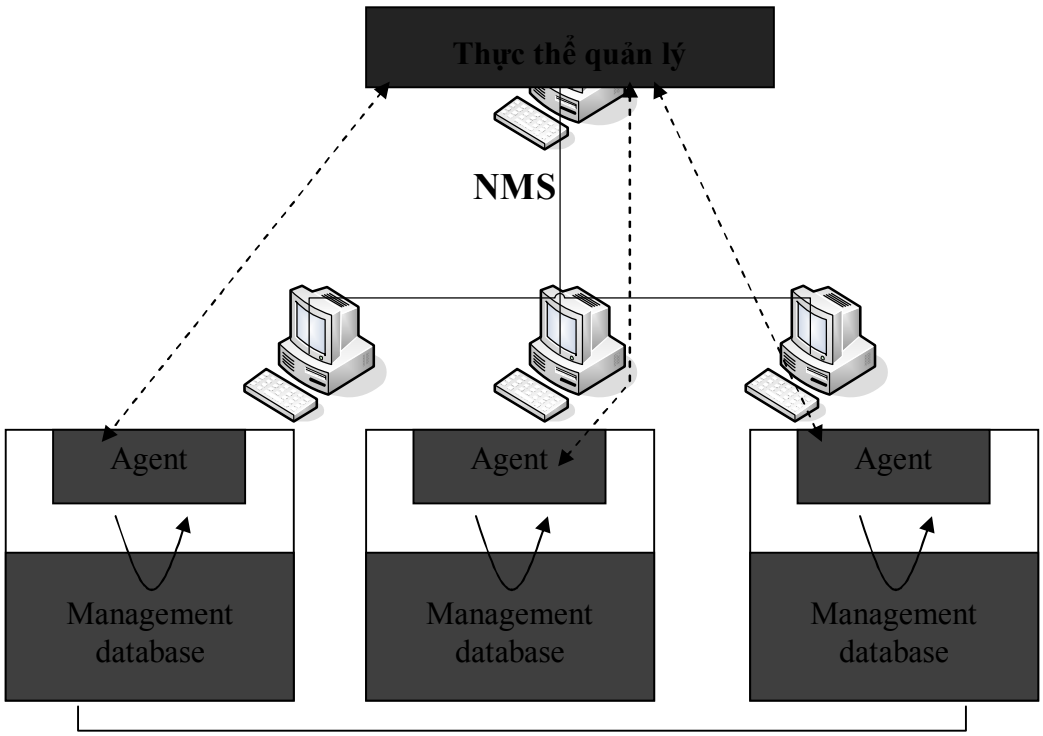
Thực thi các ứng dụng giám sát và điều khiển các thiết bị điều khiển. Qui mô về tài nguyên xử lý và bộ nhớ được yêu cầu cho quản lý mạng được cung cấp bởi NMS. Một hay nhiều NMS phải tồn tại trên bất kỳ mạng được quản lý nào.

- *Manager devices:*

Là các thiết bị được quản lý hay là các node mạng chứa một SNMP agent và cư ngụ trên một mạng được quản lý. Các thiết bị được quản lý thu thập và lưu giữ thông tin quản lý và làm cho thông tin này khả dụng đối với các NMS thông qua SNMP. Các thiết bị quản lý mạng, đôi khi còn gọi là các phần tử quản lý mạng, có thể là các router, các access server, các switch, các bridge, các hub, các computer host hay các máy in.

- *Agent:*

Các agent là các modul phần mềm quản lý mạng cư ngụ trong chính các thiết bị được quản lý. Một agent có trị thức cục bộ về thông tin quản lý mạng và thông dịch thông tin này sang dạng thức thích nghi với SNMP.



Hình 2.3: Mạng được quản lý theo SNMP

- *Các SNMP Primitive*

Bao gồm get, get-next và set. Manager dùng *get* primitive để nhận một tập thông tin đơn từ một agent. Dùng get-next nếu có nhiều hơn một item, khi dữ liệu manager cần nhận từ agent chứa đựng nhiều hơn một item, primitive này được dung để khôi phục chuỗi dữ liệu. Bạn có thể dụng set khi bạn muốn đặt một giá trị cụ thể. Manager có thể dùng primitive này để yêu cầu agent chạy trên thiết bị từ xa đặt một biến cụ thể cho giá trị hiện tại. Có hai primitive điều khiển mà responder (manager) dùng để trả lời lại đó là: get-response và trap. Một được dùng trong việc trả lời các yêu cầu trực tiếp (get-response) và một là asynchronous response nhằm thu các sự chú ý của các requester(trap). Mặc dù các sự trao đổi SNMP thường được khởi tạo bởi phần mềm manager, primitive này cũng có thể được sử dụng khi agent cần thông báo cho manager các sự kiện quan trọng, điều này thường được thông báo như một trap được gửi bởi agent đến NMS.

- *Management Information Base (MIB)*

Loại dữ liệu agent và manager trao đổi được xác định bởi một database gọi là MIB. MIB là một nơi chứa thông tin ảo. Chú ý rằng nó là một cơ sở dữ liệu nhỏ và được đặt tại agent. Thông tin được thu thập bởi agent được lưu trữ trong MIB.

Cốt lõi của SNMP là một tập hợp đơn giản các hoạt động giúp nhà quản trị mạng có thể quản lý, thay đổi trạng thái của mạng. Ví dụ chúng ta có thể dùng SNMP để tắt một interface nào đó trên router của mình, theo dõi hoạt động của card Ethernet, hoặc kiểm soát nhiệt độ trên switch và cảnh báo khi nhiệt độ quá cao.

SNMP thường tích hợp vào trong router, nhưng khác với SGMP (Simple Gateway Management Protocol) được dùng chủ yếu cho các router Internet, SNMP có thể dùng để quản lý các hệ thống Unix, Window, máy in, nguồn điện… Nói chung, tất cả các thiết bị có thể chạy các phần mềm cho phép lấy được thông tin SNMP đều có thể quản lý được. Không chỉ các thiết bị vật lý mới quản lý được mà cả những phần mềm như web server, database.

Một hướng khác của quản trị mạng là theo dõi hoạt động mạng, có nghĩa là theo dõi toàn bộ một mạng trái với theo dõi các router, host, hay các thiết bị riêng lẻ. RMON (Remote Network Monitoring) có thể giúp ta hiểu làm sao một mạng có thể tự hoạt động, làm sao các thiết bị riêng lẻ trong một mạng có thể hoạt động đồng bộ trong mạng đó. IETF là tổ chức đã đưa ra chuẩn SNMP thông qua các RFC.

- SNMP version 1: chuẩn của giao thức SNMP được định nghĩa trong RFC 1157 và là một chuẩn đầy đủ của IETF. Vấn đề bảo mật của SNMPv1 dựa trên nguyên tắc cộng đồng, không có nhiều password, chuỗi văn bản thuần và cho phép bất kỳ một ứng dụng nào đó dựa trên SNMP có

thể hiểu các hiểu các chuỗi này để có thể truy cập vào các thiết bị quản lý. Có 3 tiêu chuẩn trong: read-only, read-write và trap.

- SNMP version 2: phiên bản này dựa trên các chuỗi "community". Do đó phiên bản này được gọi là SNMPv2c, được định nghĩa trong RFC 1905, 1906, 1907 và đây chỉ là bản thử nghiệm của IETF. Mặc dù chỉ là thử nghiệm nhưng nhiều nhà sản xuất đã đưa nó vào thực nghiệm.

- SNMP version 3: là phiên bản tiếp theo được IETF đưa ra bản đầy đủ (phiên bản gần đây của SNMP), đóng vai trò an ninh cao trong quản trị mạng và đóng vai trò mạnh trong vấn đề thẩm quyền, quản lý kênh truyền riêng giữa các thực thể. Nó được khuyến nghị làm bản chuẩn, được định nghĩa trong RFC 1905, RFC 1906, RFC 1907, RFC 2271 RFC 2571, RFC 2572, RFC 2573, RFC 2574 và RFC 2575. Nó hỗ trợ các loại truyền thông riêng tư và có xác nhận giữa các thực thể.

2.1.2. SNMP Version 3

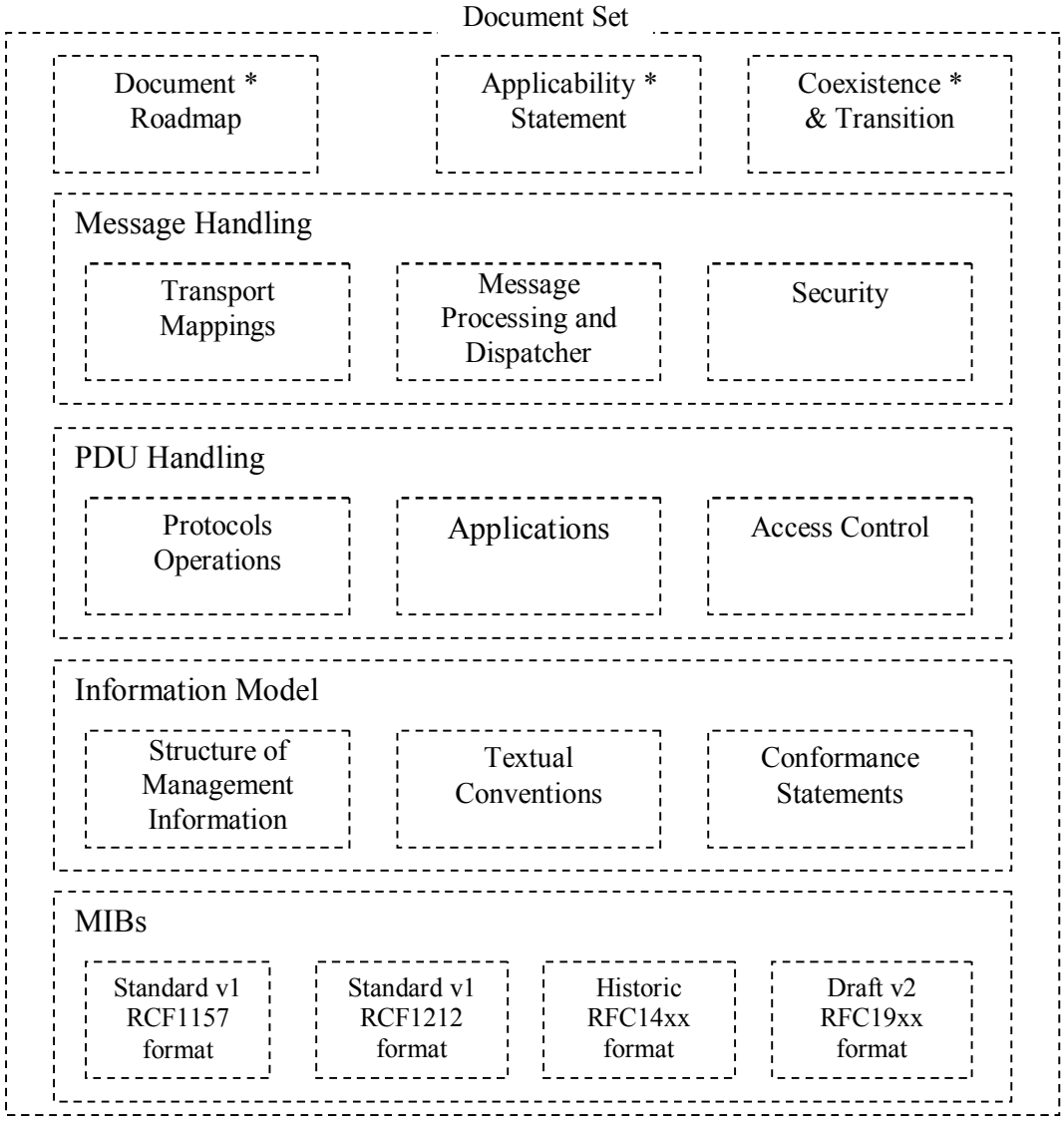
Bảo mật là vấn đề yếu kém nhất kể từ khi SNMP ra đời. Vấn đề xác thực trong SNMPv1 và SNMPv2 không gì hơn ngoài password trong clear-text giữa một máy quản trị manager và một agent. Chúng ta có thể nhận thấy vấn đề password trong clear-text thự sự là không an toàn, nó hoàn toàn có thể bị đánh cắp, truy lần lại và làm sập hệ thống mạng.

Trong SNMP Version 3 thì vấn đề bảo mật đã được quan tâm và đảm bảo an ninh hơn đối với SNMPv1 và SNMPv2. Vấn đề chính của SNMPv3 là an ninh địa chỉ, không có sự thay đổi về giao thức, không đổi mới quá trình hoạt động. SNMPv3 tích hợp tất cả các hoạt động của SNMPv1 và v2.

Tuy nhiên SNMPv3 không thay đổi đối với giao thức ngẫu nhiên từ việc bổ sung thêm các bảng mã mật. Nó được phát triển để tạo ra các khái niệm, kí hiệu mới. Thay đổi quan trọng nhất trong SNMPv3 đó là đã giải thích được ý niệm mơ hồ về manager và agent, cả manager và agent đều được gọi chung là các thực thể SNMP. Mỗi một thực thể là một SNMP

engine và sẽ có một hoặc nhiều ứng dụng chạy trên đó. Khái niệm mới này là quan trọng bởi vì chúng đã chỉ ra một cách đúng đắn nhất về kiến trúc tuyệt đối của một tập hợp các thông báo. Kiến trúc giúp tách rời các mẩu của hệ thống SNMP trong vấn đề thi hành việc bảo mật. SNMPv3 thêm vào các đặc điểm bảo mật so với SNMPv2 và SNMPv2c là: xác thực và mã hóa.

SNMPv3 sử dụng MD5 và SHA để tạo ra các giá trị hash cho từng thông điệp snmp. Thao tác này giúp cho phép xác thực các đầu cuối cũng như là ngăn ngừa thay đổi dữ liệu và các kiểu tấn công. Thêm vào đó, các phần mềm quản trị SNMPv3 và các agent có thể dùng DES để mã hóa gói tin, cho phép bảo mật tốt hơn. SNMPv3 đề nghị trong tương lai sẽ hỗ trợ

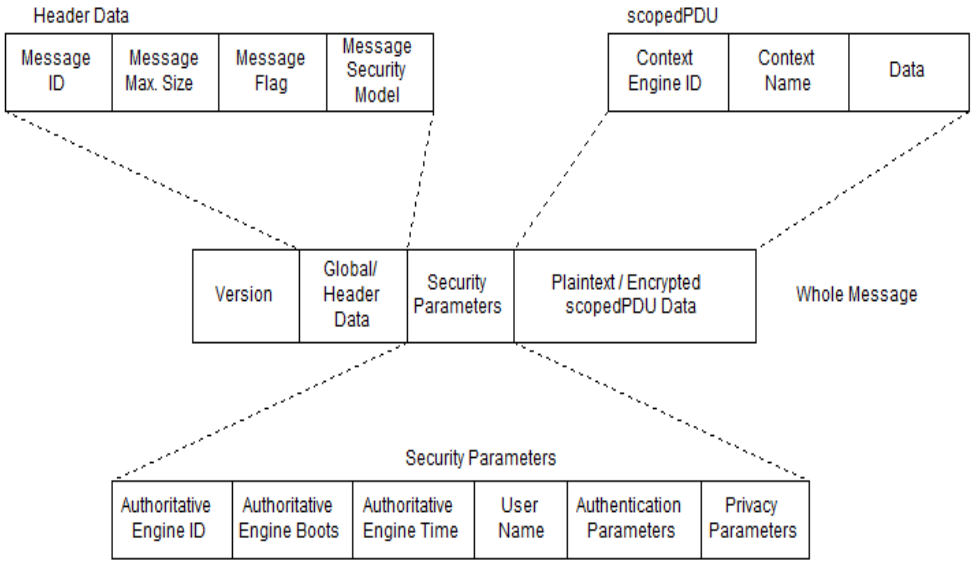


Hình 2.4 : Tổng quan kiến trúc SNMPv3

- Bảo mật trong SNMPv3

Trong một số môi trường đòi hỏi sự tác động của giao thức an ninh. Thông thường mức độ bảo mật ứng dụng ở hai giai đoạn khác nhau đó là:

- Trong quá trình truyền/nhận gói tin.
- Quá trình xử lý nội dung gói tin.

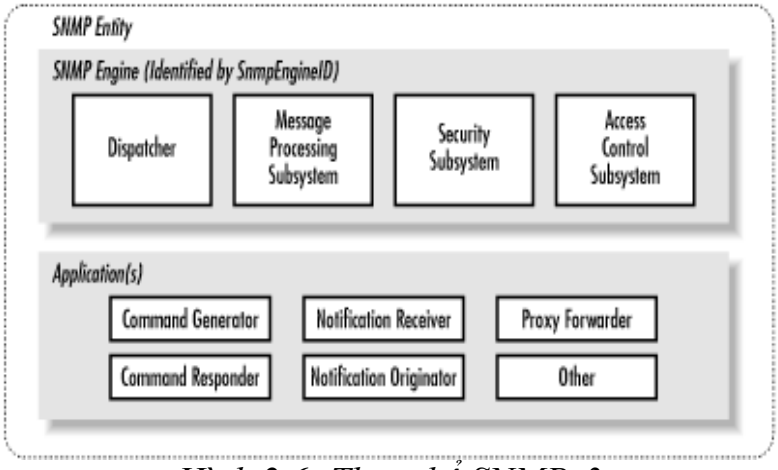


Hình 2.5: Khuôn dạng Message của SNMPv3

Trong giao thức SNMP mức độ bảo mật được ứng dụng ở mức Security Model, giao thức sử dụng trong đó và nó được định nghĩa bởi modul MIB trong suốt quá trình sử lý và cho phép cấu hình remote message-level thông qua mật khẩu.

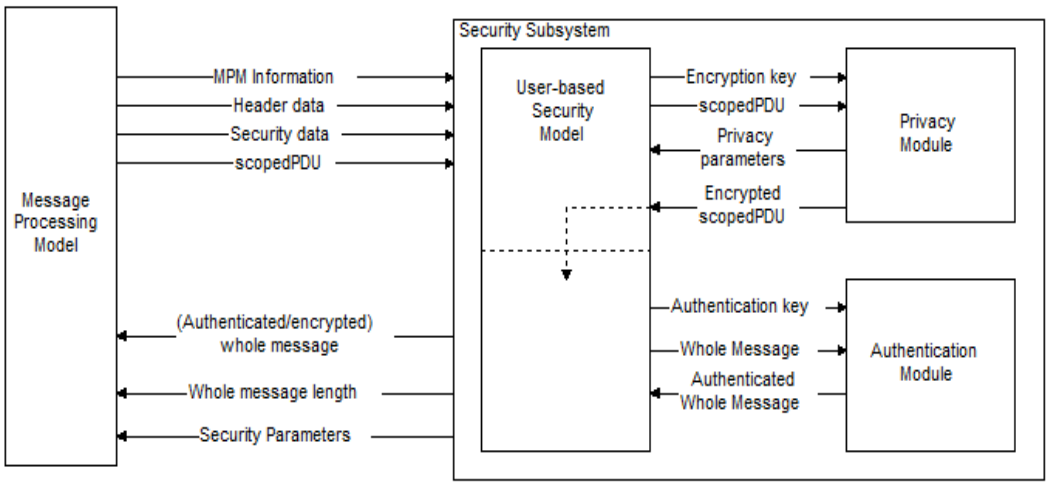
Một phần mềm SNMP phải được hỗ trợ đồng thời bởi nhiều mô hình bảo mật.

Mô hình bảo mật xác định giao thức bảo mật sử dụng để cung cấp dịch vụ bảo mật như là xác thực và bảo mật. Giao thức bảo mật được định nghĩa bởi máy xử lý và MIB dữ liệu để cung cấp dịch vụ bảo mật.



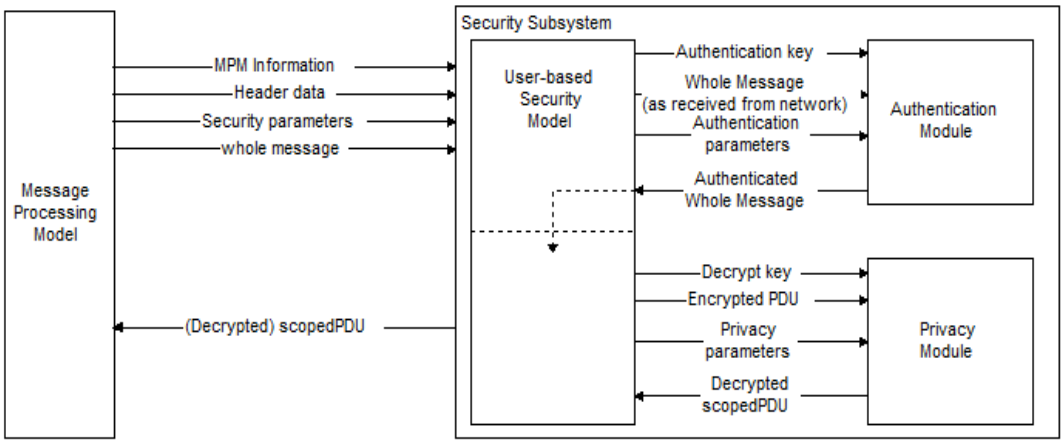
Hình 2.6: Thực thể SNMPv3

Mô hình bảo mật cơ bản tại các User (máy trạm) chủ yếu dựa trên vấn đề về cơ sở tên (name) truyền thống của User và các giao tiếp về lý thuyết USM cơ bản. Trong đó dịch vụ USM liên quan chủ yếu đến xác thực các MessageOutgoing và MessageIncoming



Hình 2.7: Dịch vụ xác thức đối với Message Outgoing

Chức năng chính của USM là chạy modul riêng w/ thông qua key và scopedPDU. Modul này trả về các biến riêng và mã hóa các scopedPDU, sau đó USM gọi chức năng xử lý w/ xác thực khóa và không làm suy chuyển các mesage, cuối cùng sẽ trả về một message đầy đủ và đã được xác thực.



Hình 2.8: Dịch vụ xác thực đối với Message Incoming

Quá trình xử lý đối với Message Incoming là hoàn toàn ngược với quá trình xử lý Message Outgoing, key cần xác thực bước đầu được đưa vào quá trình xử lý bởi chức năng xác thực kết hợp với việc giải mã thông qua modul riêng.

Modul Privacy liên quan đến một số vấn đề sau:

- Được sử dụng để mã hóa và giải mã trong phạm vi các PDU thông qua ID, name và PDU.

- Giao thức mã hóa được sử dụng bằng cách chia ra thành các đoạn (CBC – Cipher Block Chaining) sau đó sử dụng thuật toán mã hóa DES.

- Mã hóa khóa bí mật (user password) và giá trị đúng (timeliness).

- Biến riêng đó chính là giá trị *salt* (giá trị độc nhất trong CBC-DES).

Ở đây khóa sử dụng để xác thực là khóa bí mật, có sử dụng thuật toán MD5 hoặc SHA-1 để cho ta khóa xác thực ở dạng *digest2*. Khóa *degest2* thu được thông qua một số bước như sau:

- Ban đầu *digest0* được tạo ra bởi việc lặp lại password cho đến khi có dạng 2^{20} octecs.

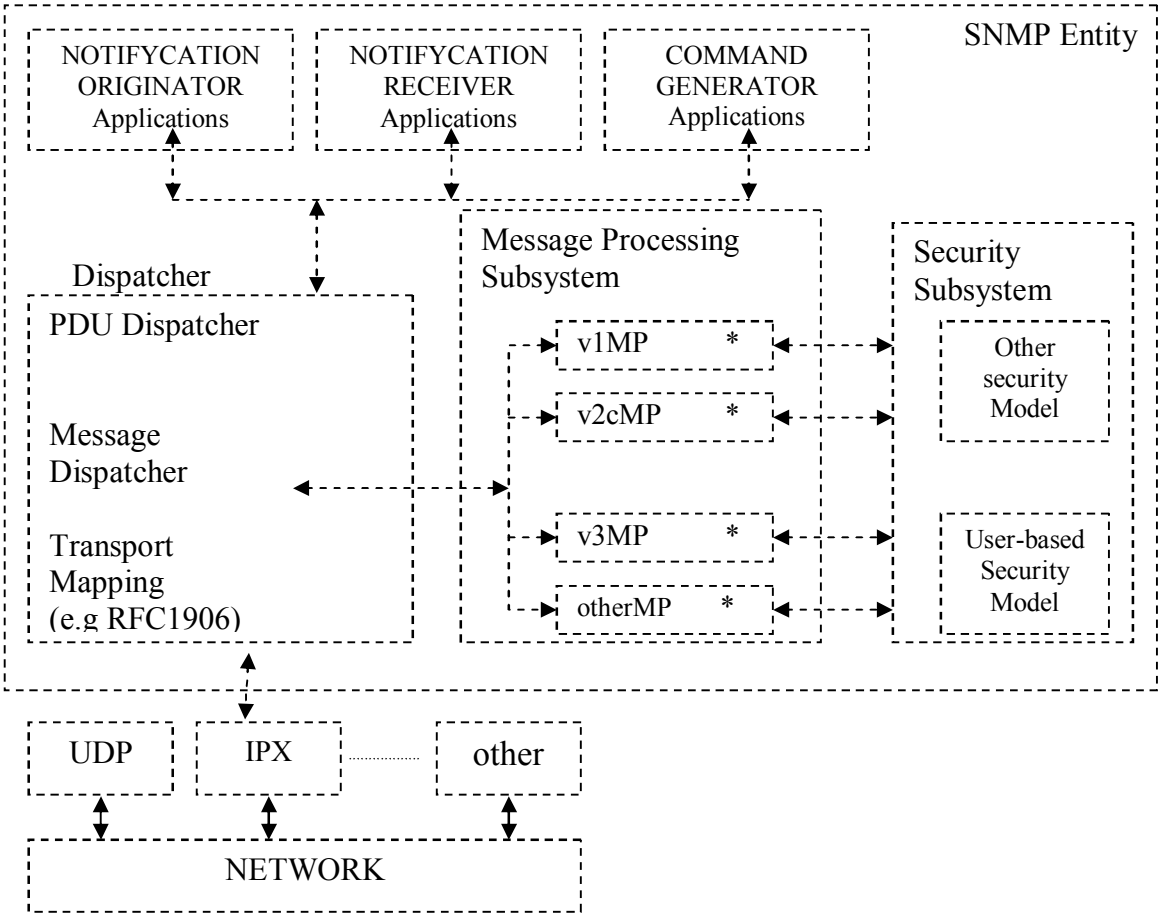
- Tiếp theo *digest1* tạo ra bởi việc sử dụng thuật toán MD5 hoặc SHA-1 đối với digest0.

- Cuối cùng *digest2* thu được từ sự kết hợp của SNMP engine ID và khóa *digest1* thông qua hàm băm.

2.1.3. Hoạt động của SNMP:

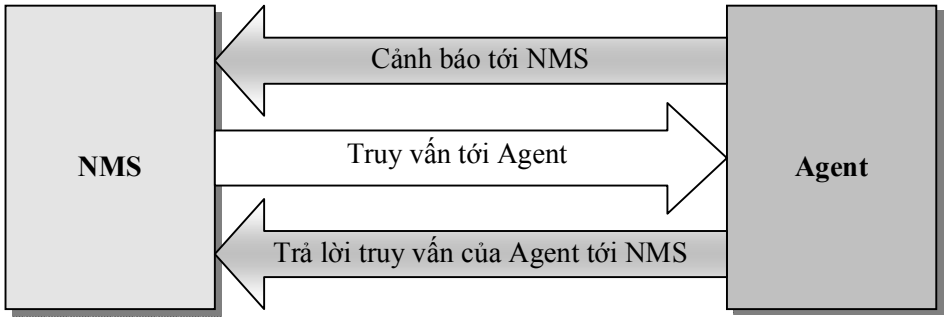
Trong SNMP có 3 vấn đề cần quan tâm: máy quản lý, agent và MIB. MIB là cơ sở dữ liệu dùng phục vụ cho Manager và Agent.

Máy quản lý là một server có chạy các chương trình có thể thực hiện một số chức năng quản lý mạng. Máy quản lý náy có thể xem như là NMS. NMS có khả năng thăm dò và thu thập các cảnh báo từ các agent trong mạng. Các cảnh báo của agent là cách mà agent báo với NMS khi có sự cố xảy ra. Cảnh báo của trạm được gửi một cách không đồng bộ, không nằm trong việc trả lời truy vấn của NMS. NMS dựa trên các thông tin trả lời của agent để có các phương án giúp mạng hoạt động hiệu quả hơn. Ví dụ khi đường dây kết nối tới Internet bị giảm băng thông nghiêm trọng, router sẽ gửi một thông tin cảnh báo tới NMS. NMS sẽ có một số hành động, ít nhất là lưu lại giúp ta có thể biết việc gì đã xảy ra. Các hành động này của NMS được cài đặt trước.



Hình 2.9: SNMP manager truyền thống

Agent là một phần trong các chương trình chạy trên các thiết bị mạng cần quản lý. Nó có thể là một chương trình độc lập như các deamon trong Unix, hoặc được tích hợp vào hệ điều hành như IOS của Cisco trên router. Ngày nay, đa số các thiết bị hoạt động tới lớp IP được cài đặt SMNP agent. Các nhà sản xuất ngày càng muốn phát triển các ứng dụng của agent trong các sản phẩm của họ công việc của người quản lý hệ thống hay quản trị mạng đơn giản hơn. Các agent cung cấp thông tin cho NMS bằng cách lưu trữ các hoạt động khác nhau của thiết bị. Một số thiết bị thường gửi một thông báo "tất cả đều bình thường" khi nó chuyển từ một trạng thái xấu sang một trạng thái tốt. Điều này giúp xác định khi nào một tình trạng có vấn đề được giải quyết.



Hình 2.10: Mối quan hệ giữa NMS và agent

Không có sự hạn chế nào khi NMS gửi một câu truy vấn đồng thời agent gửi một cảnh báo. MIB có thể xem như là một cơ sở dữ liệu của các đối tượng quản lý mà agent lưu trữ được. Bất kỳ thông tin nào mà NMS có thể truy cập được đều được định nghĩa trong MIB. Một agent có thể có nhiều MIB nhưng tất cả các agent đều có một loại MIB gọi là MIB-II được định nghĩa trong RFC 1213. MIB-I là bản gốc của MIB nhưng ít dùng khi MIB-II được đưa ra. Bất kỳ thiết bị nào hổ trợ SNMP đều phải hỗ trợ MIB-II. MIB-II định nghĩa các tham số như tình trạng của interface (tốc độ của interface, MTU, các octet gửi, các octet nhận...) hoặc các tham số gắn liền với hệ thống (định vị hệ thống, thông tin liên lạc với hệ thống,...). Mục đích

chính của MIB-II là cung cấp các thông tin quản lý theo TCP/IP. Có nhiều kiểu MIB giúp quản lý cho các mục đích khác nhau:

- ATM MIB (RFC 2515)
- Frame Relay DTE Interface Type MIB (RFC 2115)
- BGP Version 4 MIB (RFC 1657)
- RDBMS MIB (RFC 1697)
- RADIUS Authentication Server MIB (RFC 2619)
- Mail Monitoring MIB (RFC 2249)
- DNS Server MIB (RFC 1611)

Nhưng nhà sản xuất cũng như người dùng có thể định nghĩa các biến cơ sở dữ liệu riêng cho họ trong từng tình huống quản lý của họ. Quản lý tài nguyên Host cũng là một phần quan trọng của quản lý mạng. Trước đây, sự khác nhau giữa quản lý hệ thống kiểu cũ và quản lý mạng không được xác định, nhưng bây giờ đã được phân biệt rõ ràng. RFC 2790 đưa ra tài nguyên về Host với định nghĩa tập hợp cá đối tượng cần quản lý trong hệ thống Unix và Window. Một số đối tượng đó là: dung lượng đĩa, số user của hệ thống, số tiến trình đang chạy của hệ thống và các phần mềm đã cài vào hệ thống. Trong một thế giới thương mại điện tử, các dịch vụ như web ngày càng trở nên phổ biến nên việc đảm bảo cho các server hoạt động tốt là việc hết sức quan trọng RMON hay còn gọi là RMONv1 được định nghĩa trong RFC 2819. RMONv1 cung cấp cho NMS các thông tin dạng gói tin về các thực thể trong LAN hay WAN. RMONv2 được xây dựng trên RMONv1 bởi những nhà cung cấp mạng và cung cấp thông tin ở lớp ứng dụng. Thông tin có thể thu được bằng nhiều cách. Một cách trong đó là đặt một bộ phận thăm dò của RMON trên mỗi phân đoạn mạng muốn theo dõi. RMON MIB được thiết kế để các RMON có thể chạy khi không kết nối logic giữa NMS và agent, có thể lấy được thông tin mà không cần chờ truy vấn của NMS. Sau đó, khi NMS muốn truy vấn, RMON sẽ trả lời bằng các thông tin thu thập

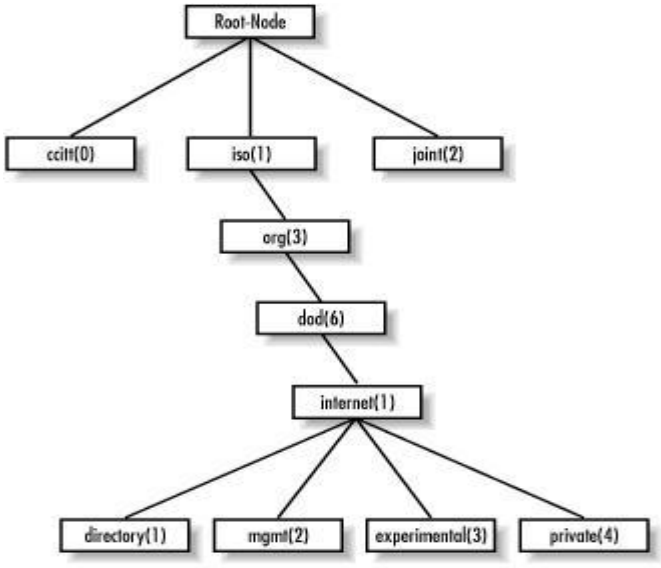
được. Một đặc tính khác là ta có thể đặt ngưỡng cho một loại lỗi nào đó, và khi lỗi vượt quá ngưỡng đặt ra, RMON gửi một cảnh báo cho NMS.

- SMI: (SMI - The Structure of Management Information) cung cấp cho chúng ta cách định nghĩa, lưu trữ các đối tượng quản lý và các thuộc tính của chúng. SMI đơn giản gồm có 3 đặc tính sau:

+ Name hay OID (object identifier): định nghĩa tên của đối tượng. Tên thường ở 2 dạng; số hay các chữ có ý nghĩa nào đó về đối tượng. Trong dạng này hay dạng kia, tên thường khó nhớ hay bất tiện.

+ Kiểu và cú pháp: Kiểu dữ liệu của object cần quản lý được định nghĩa trong ASN.1 (Abstract Syntax Notation One). ASN.1 chỉ ra cách dữ liệu được biểu diễn và truyền đi giữa máy quản lý và agent. Các thông tin mà ASN.1 thông báo là độc lập với hệ điều hành. Điều này giúp một máy chạy WindowNT có thể liên lạc với một máy chạy Sun SPARC dễ dàng.

+ Mã hóa: mã hóa các đối tượng quản lý thành các chuỗi dùng BER (Basic Encoding Rules). BER xây dựng cách mã hóa và giải mã để truyền các đối tượng qua các môi trường truyền như Ethernet. Tên hay OID được tổ chức theo dạng cây. Tên của một đối tượng được thành lập từ một dãy các số nguyên hay chữ dựa theo các nút trên cây, phân cách nhau bởi dấu chấm.



Hình 2.11: Cây đối tượng nguồn

theo mô hình cây trên ta có OID của nhánh internet:

internet OBJECT IDENTIFIER ::= {iso org(3) dod(6) 1}

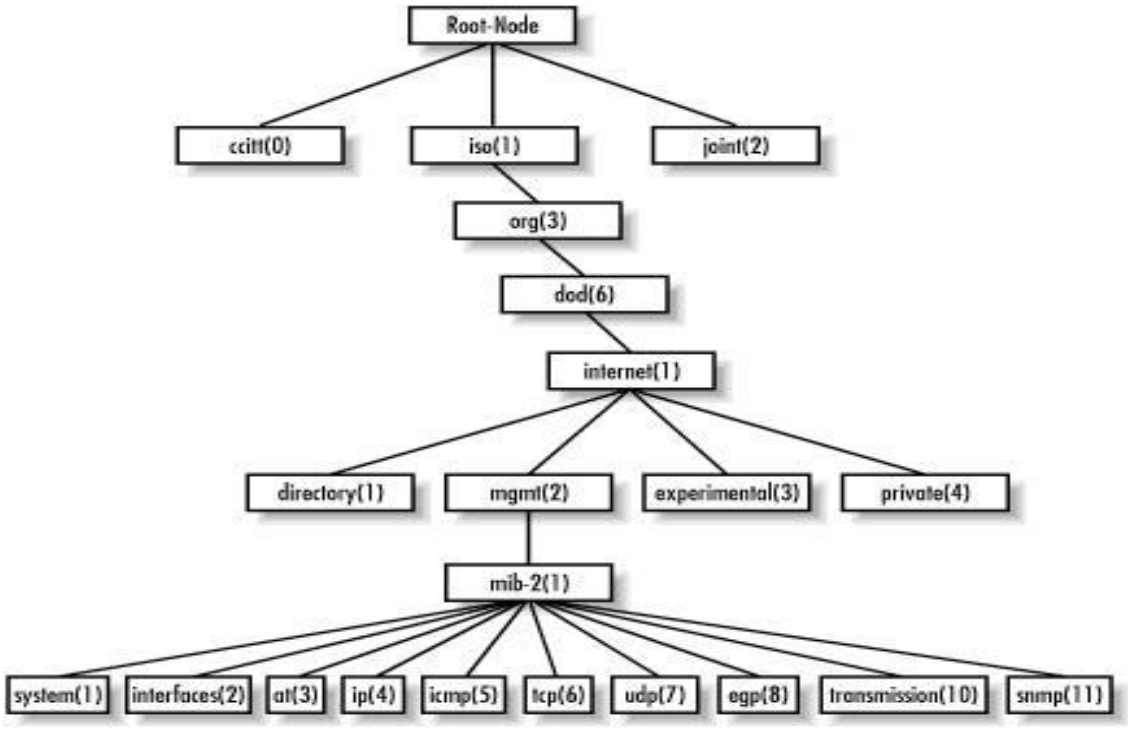
directory OBJECT IDENTIFIER ::= {internet 1}

mgmt OBJECT IDENTIFIER ::= {internet 2}

experimental OBJECT IDENTIFIER ::= {internet 3}

private OBJECT IDENTIFIER ::= {internet 4}

Trong mô hình trên, MIB-II thuộc nhánh mgmt:



Hình 2.12: Cây đối tượng kế thừa

MIB-II có 10 nhánh con được định nghĩa trong RFC 1213, kế thừa từ MIB-I trong RFC 1066. Mỗi nhánh có một chức năng riêng:

- **system** (1.3.6.1.2.1.1): Định nghĩa một danh sách các đối tượng gắn liền với hoạt động của hệ thống như: thời gian hệ thống khởi động tới bây giờ, thông tin liên lạc của hệ thống và tên của hệ thống.

- **interfaces** (1.3.6.1.2.1.2): Lưu giữ trạng thái của các interface trên một thực thể quản lý. Theo dõi một interface "up" hoặc "down", lưu lại các octet gửi và nhận, octet lỗi hay bị hủy bỏ.

- **at** (1.3.6.1.2.1.3): Nhóm at (address translation) bị phản đối, nó chỉ cung cấp khả năng tương thích ngược. Nhóm này được bỏ từ MIB-III trở đi.

- **ip** (1.3.6.1.2.1.4): Lưu giữ nhiều thông tin liên quan tới giao thức IP, trong đó có phần định tuyến IP.

- **icmp** (1.3.6.1.2.1.5): Lưu các thông tin như gói ICMP lỗi, hủy.

- **tcp** (1.3.6.1.2.1.6): Lưu các thông tin khác dành riêng cho trạng thái các kết nối TCP như: đóng, lắng nghe, báo gửi…

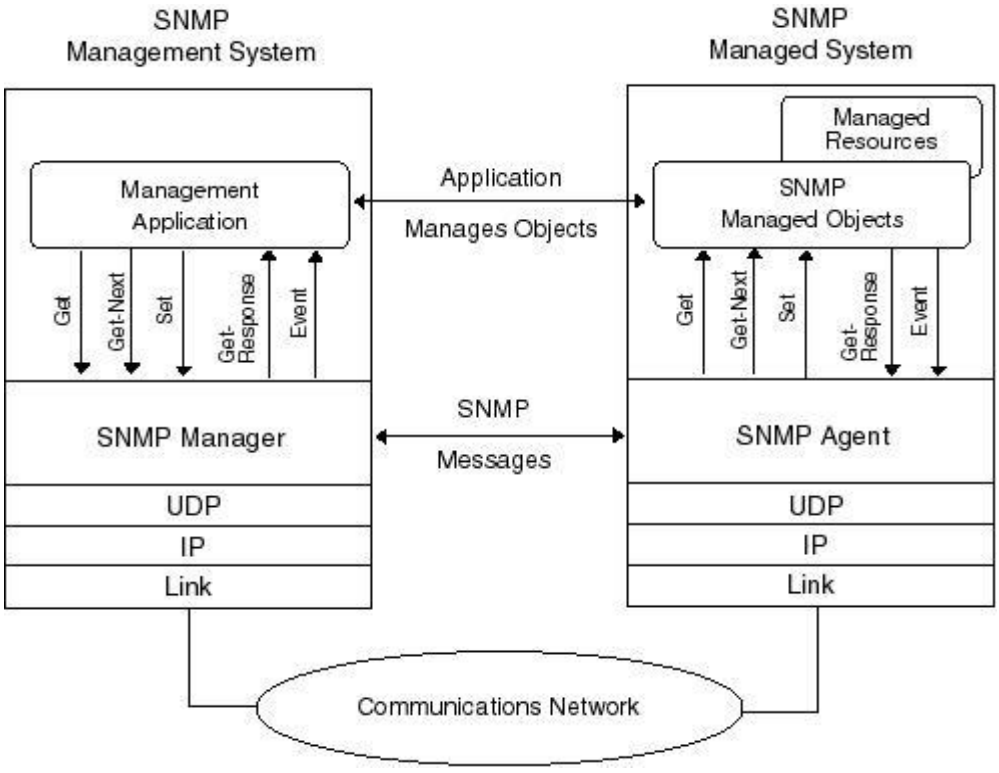
- **udp** (1.3.6.1.2.1.7): Tập hợp các thông tin thống kê cho UDP, các đơn vị dữ liệu vào và ra, …

- **egp** (1.3.6.1.2.1.8): Lưu các tham số về EGP và bảng EGP lân cận.

- **Transmission** (1.3.6.1.2.1.10): Không có đối tượng nào trong nhóm này, nhưng nó định nghĩa các môi trường đặc biệt của MIB.

- **snmp** (1.3.6.1.2.1.11): Đo lường sự thực thi của SNMP trên các thực thể quản lý và lưu các thông tin như số các gói SNMP nhận và gửi.

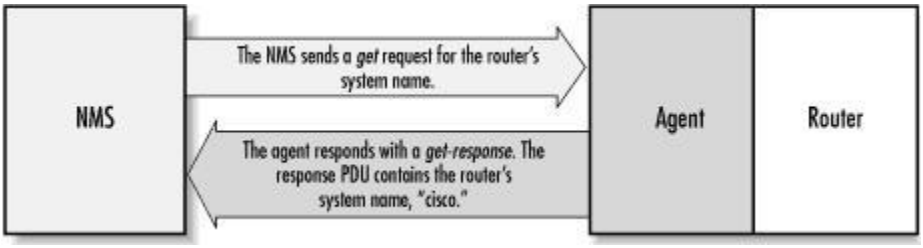
Hoạt động của SNMP theo mô hình sau:



Hình 2.13: Hoạt động của SNMP

- get
- get-next
- get-bulk (cho SNMPv2 và SNMPv3)
- set
- get-response
- trap (cảnh báo)
- notification (cho SNMPv2 và SNMPv3)
- inform (cho SNMPv2 và SNMPv3)
- report (cho SNMPv2 và SNMPv3)

- "get": "get" được gửi từ NMS yêu cầu tới agent. Agent nhận yêu cầu và xử lý với khả năng tốt nhất có thể. Nếu một thiết bị nào đó đang bận tải nặng, như router, nó không có khả năng trả lời yêu cầu nên nó sẽ hủy lời yêu cầu này. Nếu agent tập hợp đủ thông tin cần thiết cho lời yêu cầu, nó gửi lại cho NMS một "get-response":



Hình 2.14: Hoạt động của lệnh "get" trong giao thức SNMP

Để agent hiểu được NMS cần tìm thông tin gì, nó dựa vào một mục trong "get" là "variable binding" hay varbind. Varbind là một danh sách các đối tượng của MIB mà NMS muốn lấy từ agent. Agent hiểu câu hỏi theo dạng: OID=value để tìm thông tin trả lời. Câu hỏi truy vấn cho trường hợp trong hình vẽ trên:

$ snmpget cisco.ora.com public .1.3.6.1.2.1.1.6.0 system.sysLocation.0 = ""

Đây là một câu lệnh "snmpget" trên Unix. "cisco.ora.com" là tên của thiết bị, "public" là chuỗi chỉ đây là yêu cầu chỉ đọc (read-only), ".1.3.6.1.2.1.1.6.0" là OID. ".1.3.6.1.2.1.1" chỉ tới nhóm "system" trong

MIB. “.6” chỉ tới một trường trong “system” là “sysLocation”. Trong câu lệnh này ta muốn hỏi Cisco router rằng việc định vị hệ thống đã được cài đặt chưa. Câu trả lời system.sysLocation.0 = "" tức là chưa cài đặt. Câu trả lời của “snmpget” theo dạng của varbind: OID=value. Còn phần cuối trong OID ở “snmpget”; ”.0” nằm trong quy ước của MIB. Khi hỏi một đối tượng trong MIB ta cần chỉ rõ 2 trường “x.y”, ở đây là “.6.0”. “x” là OID thực tế của đối tượng. Còn “.y” được dùng trong các đối tượng có hướng như một bảng để hiểu hàng nào của bảng, với trường hợp đối tượng vô hướng như trường hợp này “y”=“0”. Các hàng trong bảng được đánh số từ số 1 trở đi. Câu lệnh “get” hữu ích trong việc truy vấn một đối tượng riêng lẻ trong MIB. Khi muốn biết thông tin về nhiều đối tượng thì “get” tốn khá nhiều thời gian. Câu lệnh “get-next” giải quyết được vấn đề này.

- “get-next”: “get-next” đưa ra một dãy các lệnh để lấy thông tin từ một nhóm trong MIB. Agent sẽ lần lượt trả lời tất cả các đối tượng có trong câu truy vấn của “get-next” tương tự như “get”, cho đến khi nào hết các đối tượng trong dãy. Ví dụ ta dùng lệnh “snmpwalk”. “snmpwalk’ tương tự như “snmpget’ nhưng không chỉ tới một đối tượng mà chỉ tới một nhánh nào đó:

$snmpwalk cisco.ora.com public system system.sysDescr.0 = "Cisco Internetwork Operating System Software ..IOS (tm) 2500 Software (C2500-I-L), Version 11.2(5), RELEASE SOFTWARE (fc1)..Copyright (c) 1986-1997 by cisco Systems, Inc...

Compiled Mon 31-Mar-97 19:53 by ckralik"

system.sysObjectID.0 = OID: enterprises.9.1.19

system.sysUpTime.0 = Timeticks: (27210723) 3 days, 3:35:07.23

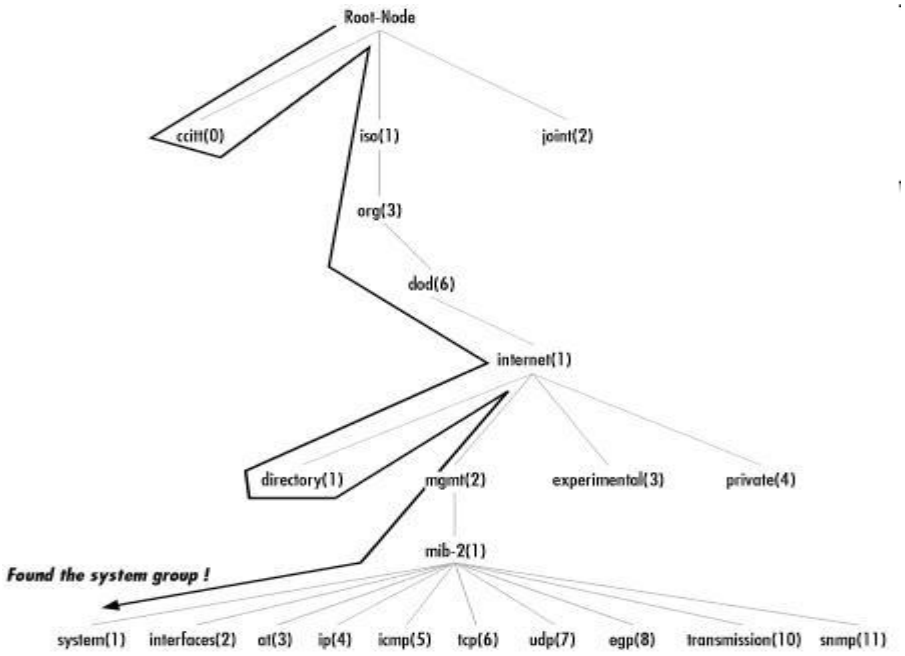
system.sysContact.0 = ""

system.sysName.0 = "cisco.ora.com"

system.sysLocation.0 = ""

system.sysServices.0 = 6

Ở đây ta muốn lấy thông tin của nhóm "system", agent sẽ gửi trả toàn bộ thông tin của "system" theo yêu cầu. Quá trình tìm nhóm "system" trong MIB thực hiện theo cây từ gốc, đến một nút nếu có nhiều nhánh thì chọn nhánh tìm theo chỉ số của nhánh từ nhỏ đến lớn:



Hình 2.15: Quá trình tìm kiếm trong cây

- "get-bulk": "get-bulk" được định nghĩa trong SNMPv2. Nó cho phép lấy thông tin quản lý từ nhiều phần trong bảng. Dùng "get" có thể làm được điều này. Tuy nhiên, kích thước của câu hỏi có thể bị giới hạn bởi agent. Khi đó nếu nó không thể trả lời toàn bộ yêu cầu, nó gửi trả một thông điệp lỗi mà không có dữ liệu. Với trường hợp dùng câu lệnh "get-bulk", agent sẽ gửi càng nhiều trả lời nếu nó có thể. Do đó, việc trả lời một phần của yêu cầu là có thể xảy ra. Hai trường cần khai báo trong "get-bulk" là: "nonrepeaters" và "max-repetitions". "nonrepeaters" báo cho agent biết N đối tượng đầu tiên có thể trả lời lại như một câu lệnh "get" đơn. "mã-repeaters" báo cho agent biết cần cố gắng tăng lên tối đa M yêu cầu "get-next" cho các đối tượng còn lại:

$ snmpbulkget -v2c -B 1 3 linux.ora.com public sysDescr ifInOctets ifOutOctets

system.sysDescr.0 = "Linux linux 2.2.5-15 #3 Thu May 27 19:33:18 EDT 1999 i686"

interfaces.ifTable.ifEntry.ifInOctets.1 = 70840

interfaces.ifTable.ifEntry.ifOutOctets.1 = 70840

interfaces.ifTable.ifEntry.ifInOctets.2 = 143548020

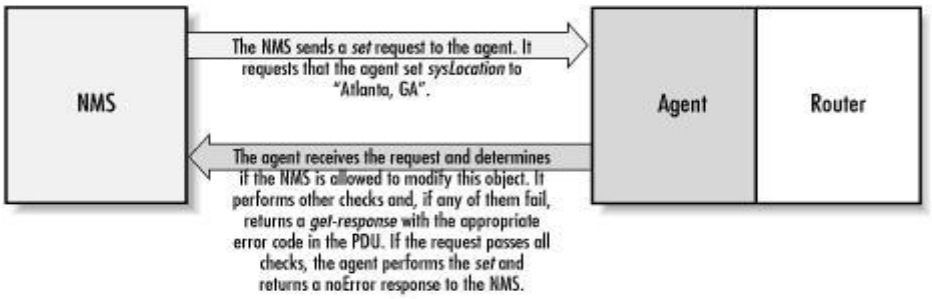
interfaces.ifTable.ifEntry.ifOutOctets.2 = 111725152

interfaces.ifTable.ifEntry.ifInOctets.3 = 0

interfaces.ifTable.ifEntry.ifOutOctets.3 = 0

Ở đây, ta hỏi về 3 varbind: sysDescr, ifInOctets, và ifOutOctets. Tổng số varbind được tính theo công thức N + (M * R) N: nonrepeater, tức số các đối tượng vô hướng M: max-repeatition R: số các đối tượng có hướng trong yêu cầu chỉ có sysDescr là vô hướng và N = 1M có thể đặt cho là 3, tức là 3 trường cho mỗi ifInOctets và ifOutOctets. Có 2 đối tượng có hướng là fInOctets và ifOutOctets ef R=2 Tổng số có 1+3*2 = 7 varbind Còn trường "–v2c" là do "get-bulk" là câu lệnh của SNMPv2 nên sử dụng "-v2c" để chỉ rằng sử dụng PDU của SNMPv2. "-B 1 3" là để đặt tham số N và M cho lệnh.

- "set": để thay đổi giá trị của một đối tượng hoặc thêm một hàng mới vào bảng. Đối tượng này cần phải được định nghĩa trong MIB là "read-write" hay "write-only". NMS có thể dùng "set' để đặt giá trị cho nhiều đối tượng cùng một lúc:



Hình 2.16: Hoạt động của Set

$ snmpget cisco.ora.com public system.sysLocation.0

system.sysLocation.0 = ""

$ snmpset cisco.ora.com private system.sysLocation.0 s "Atlanta, GA"

system.sysLocation.0 = "Atlanta, GA"

$ snmpget cisco.ora.com public system.sysLocation.0

system.sysLocation.0 = "Atlanta, GA"

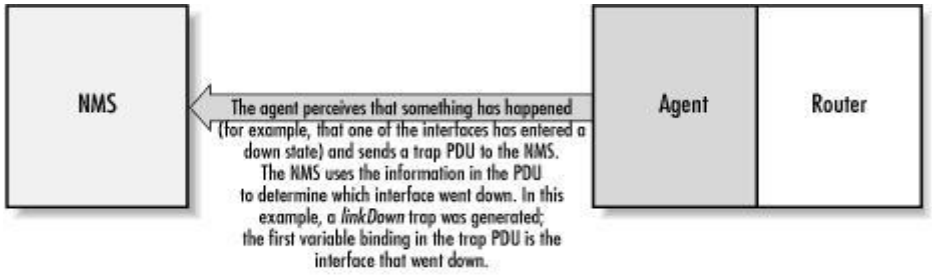
Câu lệnh đầu là dùng "get" để lấy giá trị hiện tại của "system.sysLocation". Trong câu lệnh "snmpset" các trường "cisco.ora.com" và "system.sysLocation.0" có ý nghĩa giống với "get". "private" để chỉ đối tượng "read-write", và đặt giá trị mới bằng: "s "Atlanta, GA"". "s" tức là đặt giá trị của "system.sysLocation.0" thành string, và giá trị mới là "Atlanta, GA" . Varbind này được định nghĩa trong RFC 1213 là kiểu string tối đa 255 ký tự:

sysLocation OBJECT-TYPE YNTAX DisplayString (SIZE (0..255)) ACCESS read-write TATUS mandatory ESCRIPTION The physical location of this node (e.g., 'telephone closet, 3rd floor')." ::= { system 6 } Có thể cài đặt nhiều đối tượng cùng lúc, tuy nhiên nếu có một hành động bị lỗi, toàn bộ sẽ bị hủy bỏ.

- Error Response của **"get", "get-next", "get-bulk"** và **"set":** Có nhiều loại lỗi báo lại từ agent: SNMPv1 Error Message ý nghĩa noError(0) Không có lỗi tooBig(1). Yêu cầu quá lớn để có thể dồn vào một câu trả lời noSuchName(2) OID yêu cầu không tìm thấy, tức không tồn tại ở agent badValue(3). Câu lệnh "set" dùng không đúng với các object "read-write" hay "write-only" readOnly(4) lỗi này ít dùng. Lỗi noSuchName" tương đương với lỗi này genErr(5) dùng cho tất cả các lỗi còn lại, không nằm trong các lỗi trên Các loại lỗi của SNMPv1 mang tính chất chung nhất, không rõ ràng. Do đó SNMPv2 đưa ra thêm một số loại lỗi như sau:

SNMPv2 Error Message ý nghĩa noAccess(6) lỗi khi lệnh "set" cố gắng xâm nhập vào một biến cấm xâm nhập. Khi đó, biến đó có trường "ACCESS" là "not-accessible" wrongType(7) lỗi xảy ra khi lệnh "set" đặt một kiểu dữ liệu khác với kiểu định nghĩa sẵn của đối tượng. Ví dụ khi "set" đặt giá trị kiểu string cho một đối tượng kiểu số nguyên INTEGER wrongLength(8) lỗi khi lệnh "set" đưa vào một giá trị có chiều dài lớn hơn chiều dài tối đa của đối tượng wrongEncoding(9) lỗi khi lệnh "set" sử dụng cách mã hóa khác với cách đối tượng đã định nghĩa. wrongValue(10) Một biến được đặt một giá trị mà nó không hiểu. Khi một biến theo kiểu liệt kê "enumeration" được đặt một giá trị không theo kiểu liệt kê. noCreation(11) lỗi khi cố đặt một giá trị cho một biến không tồn tại hoặc tạo một biến không có trong MIB inconsistentValue Một biến MIB ở trạng thái không nhất quán, và nó không chấp nhận bất cứ câu lệnh "set" nào. resourceUnavailable(13) Không có tài nguyên hệ thống để thực hiện lệnh "set" commitFailed(14) Đại diện cho tất cả các lỗi khi lệnh "set" thất bại undoFailed(15) Một lệnh "set" không thành công và agent không thể phục hồi lại trạng thái trước khi lệnh "set" bắt đầu thất bại. authorizationError(16) Một lệnh SNMP không được xác thực, khi một người nào đó đưa ra mật mã không đúng. notWritable(17) Một biến không chấp nhận lệnh "set" inconsistentName(18) Cố gắng đặt một giá trị, nhưng việc cố gắng thất bại vì biến đó đang ở tình trạng không nhất quán.

- SNMP Traps: Trap là cảnh báo của agent tự động gửi cho NMS để NMS biết có tình trạng xấu ở agent



Hình 2.17: Hoạt động của SNMP Trap

Khi nhận được một "trap" từ agent, NMS không trả lời lại bằng "ACK". Do đó agent không thể nào biết được là lời cảnh báo của nó có tới được NMS hay không. Khi nhận được một "trap" từ agent, nó tìm xem "trap number" để hiểu ý nghĩa của "trap" đó: Số và tên kiểu trap định nghĩa coldStart (0) Thông báo agent vừa khởi động lại. Tất cả các biến quản lý sẽ được reset, các biến kiểu "Counters" và "Gauges" được đặt về 0. "coldStart" dùng để xác định một thiết bị mới gia nhập vào mạng. Khi một thiết bị khởi động xong, nó gửi một "trap" tới NMS. Nếu địa chỉ NMS là đúng, NMS có thể nhận được và xác định xem có quản lý thiết bị đó hay không. warmStart(1) thông báo agent vừa khởi tạo lại, không có biến nào bị reset. linkDown(2) gửi đi khi một interface trên thiết bị chuyển sang trạng thái "down". linkUp(3) gửi đi khi một interface trở lại trạng thái "up". authenticationFailure(4) cảnh báo khi một người nào đó cố truy cập vào agent đó mà không được xác thực. egpNeighborLoss(5) cảnh báo một EGP lân cận bị "down" enterpriseSpecific(6) đây là một "trap" riêng, chỉ được biết bởi agent và NMS tự định nghĩa riêng chúng. NMS sử dụng phương pháp giải mã đặc biệt để hiểu được thông điệp này. "trap" được đưa ra trong MIB qua "rdbmsOutOfSpace": *rdbmsOutOfSpace TRAP-TYPE ENTERPRISE rdbmsTraps VARIABLES { rdbmsSrvInfoDiskOutOfSpaces } DESCRIPTION "An rdbmsOutOfSpace trap signifies that one of the database servers managed by this agent has been unable to allocate space for one of the databases managed by this agent. Care should be taken to avoid flooding the network with these traps." ::= 2*

- SNMP Notification: Để chuẩn hóa định dạng PDU "trap" của SNMPv1 do PDU của "get" và "set" khác nhau, SNMPv2 đưa ra NOTIFICATION-TYPE". Định dạng PDU của "NOTIFICATION-TYPE" là để nhận ra "get" và "set". *"NOTIFICATION-TYPE" được định nghĩa trong RFC 2863: linkDown NOTIFICATION-TYPE OBJECTS {ifIndex, ifAdminStatus,*

ifOperStatus} STATUS current DESCRIPTION "A linkDown trap signifies that the SNMPv2 entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links left the down state and transitioned into some other state (but not into the notPresent state). This other state is indicated by the included value of ifOperStatus." ::= { snmpTraps 3 } OID của "trap" này là 1.3.6.1.6.3.1.1.5.3, tức iso.org.dod.internet.snmpV2.snmpModules.snmpMIB.snmpMIBObjects.snmpTraps.linkDown.

- SNMP inform: SNMPv2 cung cấp cơ chế truyền thông giữa những NMS với nhau, gọi là SNMP inform. Khi một NMS gửi một SNMP inform cho một NMS khác, NMS nhận được sẽ gửi trả một ACK xác nhận sự kiện. Việc này giống với cơ chế của "get" và "set".

- SNMP report: được định nghĩa trong bản nháp của SNMPv2 nhưng không được phát triển. Sau đó được đưa vào SNMPv3 và hy vọng dùng để truyền thông giữa các hệ thống SNMP với nhau

2.2. Các giải pháp xác thực thông tin quản trị

Xác thực là một phần quan trọng trong cấu trúc an ninh quản trị mạng nói chung và của kiến trúc quản trị mạng SNMP nói riêng. Người sử dụng cần phải được xác thực để có thể truy cập vào tài nguyên của hệ thống. Sau đây là một số phương thức xác thực:

+ **PAP** (Password Authentication Protocol): Là một giao thức xác thực đơn giản dựa trên mật khẩu, sử dụng các kết nối PPP. PAP không an toàn vì thông tin xác thực không được mã hóa, kẻ tấn công có thể chặn và đọc được mật khẩu và giả danh người sử dụng để truy nhập vào mạng.

+ **CHAP** (Challenge Handshake Authentication Protocol): là giao thức xác thực có kiểm tra, cũng được sử dụng trong các kết nối PPP. Quá trình thực hiện phương thức xác thực CHAP bao gồm 3 bước

- Bên xác thực gửi thông điệp yêu cầu tới đầu bên kia (user).

- Đầu bên kia tính toán một giá trị bằng cách sử dụng hàm băm một chiều và gửi trả lại cho bên xác thực.

- Bên xác thực có thể chấp nhận xác thực nếu giá trị đó phù hợp.

Sau khi thoả thận giao thức xác thực CHAP trên liên kết PPP giữa các đầu cuối, máy chủ truy cập gửi một "chanllenge" tới người dùng từ xa. Người dùng từ xa phúc đáp lại một giá trị được tính toán sử dụng tiến trình xử lý một chiều (hash), máy chủ truy cập kiểm tra và so sánh thông tin phúc đáp với giá trị hash mà nó vừa tính được. Nếu gía trị bằng nhau, xác thực thành công, ngược lại kết nối sẽ bị huỷ bỏ.

Chap cung cấp cơ chế an toàn thông qua việc sử dụng giá trị chanllenge thay đổi, duy nhất và không thể đoán được

Nhược điểm của phương pháp xác thực này là tính khả mở kém vì nó yêu cầu quản lý một lượng lớn các thuộc tính sử dụng cho hàm băm, đặc biệt trong các mạng lớn.

+ **TACACS** (Terminal Access Controller Access-Control System): Hệ thống điều khiển truy nhập từ xa trong mô hình khách/chủ.

- User quay số tới máy chủ truy nhập từ xa.

- RAS(Remote Access Service) sử dụng giao thức TACACS/RADIUS gửi yêu cầu tới máy chủ xác thực (Authentication server).

- Máy chủ xác thực kiểm tra yêu cầu dựa vào cơ sở dữ liệu tài khoản người sử dụng.

+ **RADIUS** (Remote Authentication Dial-In User Service): Là dịch vụ xác thực truy nhập từ xa, hỗ trợ nhiều máy chủ và số lượng lớn kết nối. Mô hình khách/chủ RADIUS sử dụng một máy chủ điều khiển truy nhập (NAS – Network Access Service) để quản lý kết nối, nó cũng có chức năng như máy khách của RADIUS. Truyền thông giữa máy khách và máy chủ

RADIUS là được bảo mật, sử dụng mật khóa chia sẻ cho xác thực và mã hóa để truyền mật khẩu của người sử dụng.

+ **CA** (Certificate Authentication): Chứng thực điện tử - là một tổ chức cấp chứng chỉ số.

2.3. Giải pháp đảm bảo toàn vẹn thông tin quản trị

Các bước bảo vệ:

+ Điều khiển truy nhập (Access Control): cấp quyền truy nhập và sử dụng tài nguyên, xác thực, giám sát truy nhập.

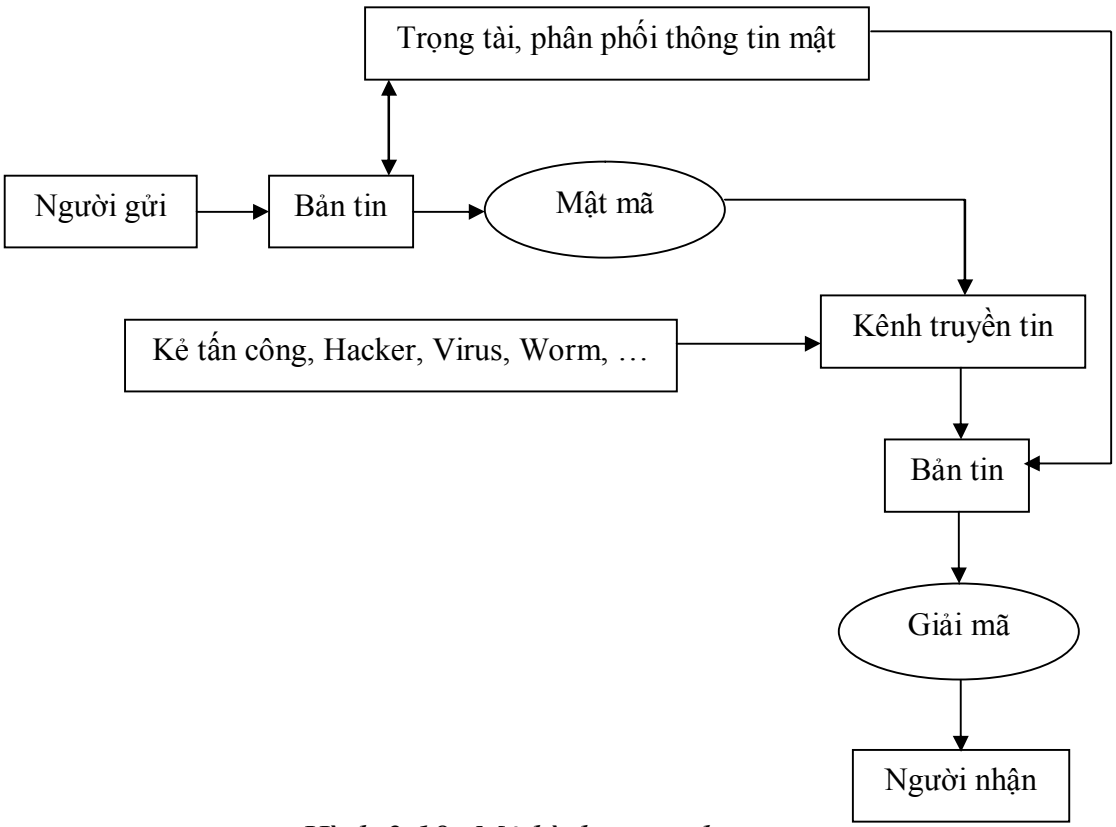
+ Giám sát hoạt động mạng.

+ Bảo mật thông tin trên mạng (mã DES, mã công khai RSA …)

+ Bảo vệ vật lý: ngăn cản truy nhập vật lý bất hợp pháp (gate keeper)

+ Kiểm soát phần mềm đưa vào mạng

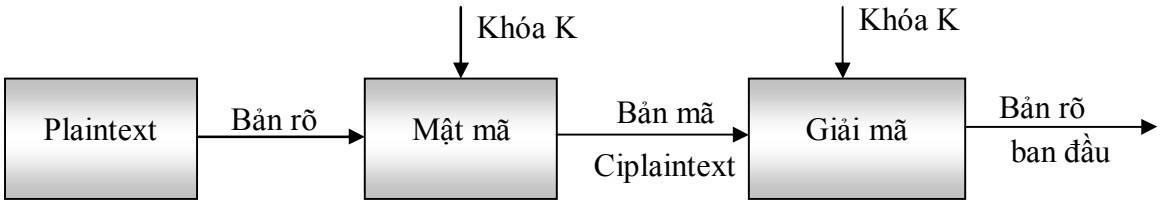
+ Bức tường lửa (Firewall) ngăn các mạng nội bộ với thế giới Internet bên ngoài.



Hình 2.18: Mô hình an ninh mạng

2.4. Giải pháp mã mật thông tin quản trị

Thuật toán Cryptography đề cập tới ngành khoa học nghiên cứu về mã hoá và giải mã thông tin. Cụ thể hơn là nghiên cứu các cách thức chuyển đổi thông tin từ dạng rõ (clear text) sang dạng mờ (cipher text) và ngược lại. Đây là một phương pháp hỗ trợ rất tốt cho trong việc chống lại những truy cập bất hợp pháp tới dữ liệu được truyền đi trên mạng, áp dụng mã hoá sẽ khiến cho nội dung thông tin được truyền đi dưới dạng mờ và không thể đọc được đối với bất kỳ ai cố tình muốn lấy thông tin đó.



Hình 2.19: Quá trình mã mật thông tin

Không phải ai hay bất kỳ ứng dụng nào cũng phải sử dụng mã hoá. Nhu cầu về sử dụng mã hoá xuất hiện khi các bên tham gia trao đổi thông tin muốn bảo vệ các tài liệu quan trọng hay gửi chúng đi một cách an toàn. Các tài liệu quan trọng có thể là: tài liệu quân sự, tài chính, kinh doanh hoặc đơn giản là một thông tin nào đó mang tính riêng tư,...

Như chúng ta đã biết, Internet hình thành và phát triển từ yêu cầu của chính phủ Mỹ nhằm phục vụ cho mục đích quân sự. Khi chúng ta tham gia trao đổi thông tin, thì Internet là môi trường không an toàn, đầy rủi ro và nguy hiểm, không có gì đảm bảo rằng thông tin mà chúng ta truyền đi không bị đọc trộm trên đường truyền. Do đó, mã hoá được áp dụng như một biện pháp nhằm giúp chúng ta tự bảo vệ chính mình cũng như những thông tin mà chúng ta gửi đi. Bên cạnh đó, mã hoá còn có những ứng dụng khác như là bảo đảm tính toàn vẹn của dữ liệu.

Theo một số tài liệu thì trước đây tính an toàn, bí mật của một thuật toán phụ thuộc vào phương thức làm việc của thuật toán đó. Nếu như tính an toàn của một thuật toán chỉ dựa vào sự bí mật của thuật toán đó thì thuật

toán đó là một thuật toán hạn chế (Restricted Algrorithm). Thuật toán này có tầm quan trọng trong lịch sử nhưng không còn phù hợp trong thời đại ngày nay. Giờ đây, nó không còn được mọi người sử dụng do mặt hạn chế của nó: mỗi khi một user rời khỏi một nhóm thì toàn bộ nhóm đó phải chuyển sang sử dụng thuật toán khác hoặc nếu người đó người trong nhóm đó tiết lộ thông tin về thuật toán hay có kẻ phát hiện ra tính bí mật của thuật toán thì coi như thuật toán đó đã bị phá vỡ, tất cả những user còn lại trong nhóm buộc phải thay đổi lại thuật toán dẫn đến mất thời gian và công sức.

Hệ thống mã hoá hiện nay đã giải quyết vấn đề trên thông qua khoá là một yếu tố có liên quan nhưng tách rời ra khỏi thuật toán mã hoá. Do các thuật toán hầu như được công khai cho nên tính an toàn của mã hoá giờ đây phụ thuộc vào khoá. Khoá này có thể là bất kì một giá trị chữ hoặc số nào. Phạm vi không gian các giá trị có thể có của khoá được gọi là Keyspace. Hai quá trình mã hoá và giải mã đều dùng đến khoá. Hiện nay, người ta phân loại thuật toán dựa trên số lượng và đặc tính của khoá được sử dụng.

Nói đến mã hoá tức là nói đến việc che dấu thông tin bằng cách sử dụng thuật toán. Che dấu ở đây không phải là làm cho thông tin biến mất mà là cách thức chuyển từ dạng tỏ sang dạng mờ. Một thuật toán là một tập hợp của các câu lệnh mà theo đó chương trình sẽ biết phải làm thế nào để xáo trộn hay phục hồi lại dữ liệu.

Giải pháp mã mật thông tin quản trị thường sử dụng là hệ mã cổ điển DES (Data Encryption Standard): DES là tổ hợp của các phương pháp thay thế, đổi chỗ. Nó chia bản tin thành các block có độ dài cố định (64 bít) và lặp lại các phép mã hóa thay thế và đổi chỗ nhiều lần cho mỗi khối.

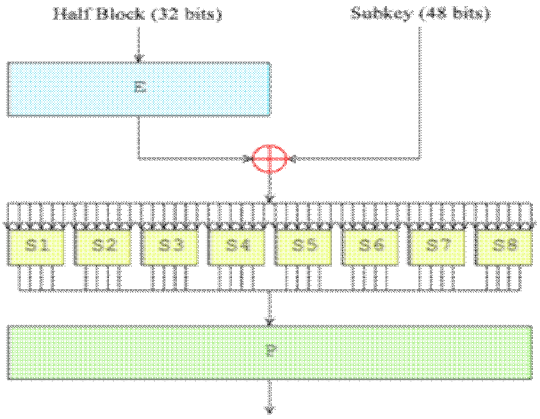
- Các phát triển tiếp của DES là:

+ IDEA (International Data Encryption Algorithm): khóa 128 bít, khối dữ liệu 64 bít.

+ RC5: khối dữ liệu và khóa sử dụng có độ dài thay đổi.

+ RC6: nâng cấp của RC5 để tăng tính bảo mật và hiệu quả.

+ AES (Advanced Encryption Standard): khối dữ liệu 128 bít, khóa 128/192/256.



Hình 2.20: Mô hình DES

2.4.1. Sơ lược mật mã đối xứng DES

Năm 1972, Viện tiêu chuẩn và công nghệ quốc gia Hoa kỳ (National Institute of Standards and Technology-NIST) đặt ra yêu cầu xây dựng một thuật toán mã hoá bảo mật thông tin với yêu cầu là dễ thực hiện, sử dụng được rộng rãi trong nhiều lĩnh vực và mức độ bảo mật cao. Năm 1974, IBM giới thiệu thuật toán Lucifer, thuật toán này đáp ứng hầu hết các yêu cầu của NIST. Sau một số sửa đổi, năm 1976, Lucifer được NIST công nhận là chuẩn quốc gia Hoa kỳ và được đổi tên thành Data Encryption Standard - DES.

DES là thuật toán mã hoá bảo mật được sử dụng rộng rãi nhất trên thế giới, thậm chí, đối với nhiều người DES và mã hoá bảo mật là đồng nghĩa với nhau. ở thời điểm DES ra đời người ta đã tính toán rằng việc phá được khoá mã DES là rất khó khăn. Cùng với sự phát triển của các loại máy tính và mạng máy tính có tốc độ tính toán rất cao, khoá mã DES có thể bị phá trong khoảng thời gian ngày càng ngắn với chi phí ngày càng thấp. Dù vậy việc này vẫn vượt xa khả năng của các hacker thông thường và mã hoá DES vẫn tiếp tục tồn tại trong nhiều lĩnh vực như ngân hàng, thương mại, thông

tin... nhiều năm nữa đặc biệt với sự ra đời của thế hệ DES mới-"Triple DES".

Kể từ khi DES ra đời, nhiều thuật toán mã hoá bảo mật khác cũng được phát triển tương tự DES hoặc dựa trên DES, một khi nắm được các nguyên tắc của DES bạn sẽ dễ dàng hiểu các thuật toán này.

Yêu cầu đặt ra nếu muốn bảo mật tốt hơn là phải tìm được một thuật toán sao cho việc thực hiện không quá phức tạp nhưng xác suất tìm ra chìa khoá bằng cách thử tất cả các trường hợp (brute-force) là rất nhỏ (số lần thử phải rất lớn).

2.4.2. Thuật toán bảo mật DES.

Về mặt khái niệm, thông thường thuật toán mã hoá DES là thuật toán mở, nghĩa là mọi người đều biết thuật toán này. Điều quan trọng nhất là chìa khoá của DES có độ dài tới 56 bit, nghĩa là số lần thử tối đa để tìm được chìa khoá lên đến 2^56, trung bình là 2^55 = 36.028.797.018.963.968 lần, một con số rất lớn.

DES được thực hiện nhờ các phép dịch, hoán vị và các phép toán logic trên các bit. Mỗi ký tự trong bức thư hay bản tin cần mã hoá được biểu diễn bởi 2 số hexa hay 8 bít. DES mã hoá từng khối 64 bít tương đương 16 số hexa. Để thực hiện việc mã hoá DES sử dụng một chìa khoá cũng dưới dạng 16 số hexa hay 64 bít tức 8 byte, nhưng các bít thứ 8 trong các byte này bị bỏ qua trong khi mã hoá vì vậy độ lớn thực tế của chìa khoá là 56 bit. Ví dụ, ta mã hoá một bản tin hexa "0123456789ABCDEF" với chìa khoá là "5A5A5A5A5A5A5A5A" thì kết quả là "72AAE3B3D6916E92". Nếu kết quả này được giải mã với cùng chìa khoá "5A5A5A5A5A5A5A5A" thì ta sẽ thu lại được đúng bản tin "0123456789ABCDEF". DES bao gồm 16 vòng, nghĩa là thuật toán chính được lặp lại 16 lần để tạo ra bản tin được mã hoá. Sau đây tôi sẽ trình bày quy trình của thuật toán DES.

2.4.2.1. Chuẩn bị chìa khoá:

Bước đầu tiên là chuyển 64 bit chìa khoá qua một bảng hoán vị gọi là Permuted Choice hay PC-1 để thu được chìa khoá mới có 56 bit.

Sau khi vệc chuẩn bị chìa khoá và dữ liệu mã hoá hoàn thành, thực hiện mã hoá bằng thuật toán DES. Đầu tiên, khối dữ liệu đầu vào 64 bit được chia thành hai nửa, L và R. L gồm 32 bit bên trái và R gồm 32 bit bên phải. Quá trình sau đây được lặp lại 16 lần tạo thành 16 vòng của DES gồm 16 cặp L[0]-L[15] và R[0]-R[15]:

Bước 1: R[r-1]- ở đây r là số vòng, bắt đầu từ 1- được lấy và cho qua bảng E (E-bit Selection Table), bảng này giống như một bảng hoán vị, có điều là một số bit được dùng hơn một lần do vậy nó sẽ mở rộng R[r-1] từ 32 bit lên 48 bit để chuẩn bị cho bước tiếp theo.

Bước 2: 48 bit R[r-1] được XOR với K[r] và được lưu trong bộ nhớ đệm, vì vậy R[r-1] không thay đổi.

Bước 3: Kết quả của bước trước lại được chia thành 8 đoạn, mỗi đoạn 6 bit, từ B[1] đến B[8]. Những đoạn này tạo thành chỉ số cho các bảng S (Substitution) được sử dụng ở bước tiếp theo. Các bảng S, là một bộ 8 bảng (S[1]-S[8]) 4 hàng, 16 cột. Các số trong bảng có độ dài 4 bit vì vậy có giá trị từ 0 đến 15.

Bước 4: Bắt đầu từ B[1], bit đầu và cuối của khối 6 bit được lấy ra và sử dụng làm chỉ số hàng của bảng S[1], nó có giá trị từ 0 đến 3, và 4 bit giữa được dùng làm chỉ số cột, từ 0 đến 15. Giá trị được chỉ đến trong bảng S được lấy ra và lưu lại. Việc này được lặp lại đối với B[2] và S[2] cho đến B[8] và S[8]. Lúc này bạn có 8 số 4 bit, khi nối lại với nhau theo thứ tự thu được sẽ tạo ra một chuỗi 32 bit.

Bước 5: Kết quả của bước trước được hoán vị bit bằng bảng hoán vị P (Permutation).

Bước 6: Kết quả thu được sau khi hoán vị được XOR với L[r-1] và chuyển vào R[r]. R[r-1] được chuyển vào L[r].

Bước 7: Lúc này bạn có L[r] và R[r] mới. Bạn tiếp tục tăng r và lặp lại các bước trên cho đến khi r= 17, đIều đó có nghĩa là 16 vòng đã được thực hiện và các chìa khoá phụ K[1]-K[16] đã được sử dụng.

Khi đã có L[16] và R[16], chúng được ghép lại với nhau theo cách chúng bị tách ra (L[16] ở bên trái và R[16] ở bên phải) thành 64 bit. 64 bit này được hoán vị để tạo ra kết quả cuối cùng là dữ liệu 64 bit đã được mã hoá.

2.4.2.2. Giải mã:

Việc giải mã dùng cùng một thuật toán như việc mã hoá. Để giải mã dữ liệu đã được mã hoá, quá trình như giống như mã hoá được lăp lại nhưng các chìa khoá phụ được dùng theo thứ tự ngược lại từ K[16] đến K[1], nghĩa là trong bước 2 của quá trình *mã hoá dữ liệu đầu vào* ở trên R[r-1] sẽ được XOR với K[17-r] chứ không phải với K[r].

- Các chế độ của DES:

Thuật toán DES mã hoá đoạn tin 64 bit thành đoạn tin mã hoá 64 bit. Nếu mỗi khối 64 bit được mã hoá một cách độc lập thì ta có chế độ mã hoá ECB (Electronic Code Book). Có hai chế độ khác của mã hoá DES là CBC (Chain Block Coding) và CFB (Cipher Feedback), nó làm cho mỗi đoạn tin mã hoá 64 bit phụ thuộc vào các đoạn tin trước đó thông qua phép toán XOR.

- Triple DES:

Triple-DES chính là DES với hai chìa khoá 56 bit. Cho một bản tin cần mã hoá, chìa khoá đầu tiên được dùng để mã hoá DES bản tin đó, kết quả thu được lại được cho qua quá trình giải mã DES nhưng với chìa khoá là chìa khoá thứ hai, bản tin sau qua đã được biến đổi bằng thuật toán DES hai lần như vậy lại được mã hoá DES với một lần nữa với chìa khoá đầu tiên để ra được bản tin mã hoá cuối cùng. Quá trình mã hoá DES ba bước này được gọi là Triple-DES.

- Ứng dụng của DES

DES thường được dùng để mã hoá bảo mật các thông tin trong quá trình truyền tin cũng như lưu trữ thông tin. Một ứng dụng quan trọng khác của DES là kiểm tra tính xác thực của mật khẩu truy nhập vào một hệ thống (hệ thống quản lý bán hàng, quản lý thiết bị viễn thông…), hay tạo và kiểm tính hợp lệ của một mã số bí mật (thẻ internet, thẻ điện thoại di động trả trước), hoặc của một thẻ thông minh (thẻ tín dụng, thẻ payphone…).

- Phá khóa DES

Năm 1998, một nhóm nghiên cứu đã chi phí 220.000USD để chế tạo một thiết bị có thể thử toàn bộ số chìa khoá DES 56 bit trong trung bình 4,5 ngày. Tháng 7 năm 1998 họ thông báo đã phá chìa khoá DES trong 56 giờ. Thiết bị này gọi là Deep Crack gồm 27 board mạch, mỗi board chứa 64 chip và có khả năng thử 90 tỷ chìa khoá trong một giây.

Tuy nhiên, việc phá khóa Triple DES là điều rất khó khăn, một chuyên gia về bảo mật đã cho rằng "Không có đủ silic trong giải ngân hà (để chế tạo chip-TG) cũng như không đủ thời gian trước khi mặt trời bị phá huỷ để phá khoá Triple DES".

Chương 3: MÔ HÌNH THỬ NGHIỆM

3.1. Lựa chọn mô hình thử nghiệm

Có thể nói trên thế giới hiện nay có rất nhiều phần mềm SNMP sử dụng cho version 1, version 2 và version 3. Mỗi phần mềm có thể phục vụ cho một hay nhiều hệ điều hành và có nhiều phiên bản với những ưu và nhược điểm khác nhau, có thể sử dụng cấu hình trên thiết bị phần cứng hỗ trợ giao thức SNMP như Router, Pix, Modem…. Do đó, để lựa chọn một phần mềm nhằm thực hiện thử nghiệm mô hình quản trị mạng SNMP là rất khó khăn. Sau thời gian nghiên cứu và tìm hiểu tôi đã chọn phần mềm PRTG Traffic Grapher là một ứng dụng trên Windows để minh họa cho giao thức SNMP - PRTG Traffic Grapher dùng để theo dõi và phân loại cách dùng băng thông.

Có thể nói với bất cứ một mạng nào thì việc giao tiếp với mạng bên ngoài đều phải thông qua một thiết bị đó chính là Modem. Với công cụ PRTG Traffic Grapher này người quản trị mạng có thể theo dõi băng thông của router ADSL thông qua giao thức SNMP. Trong khi các chương trình theo dõi băng thông dựa vào các traffic thông qua cổng card mạng của 1 máy tính trên mạng thì PRTG Traffic Grapher sẽ theo dõi traffic trực tiếp tại cổng PPP của router ADSL nên theo dõi được toàn bộ traffic IN, OUT của toàn mạng.

PRTG Traffic Grapher có phiên bản miễn phí, bản thương mại. Phiên bản PRTG Traffic Grapher miễn phí (15MB) được download theo link sau: http://www.paessler.com/download/prtg.

- Xác lập SNMP trên router ADSL

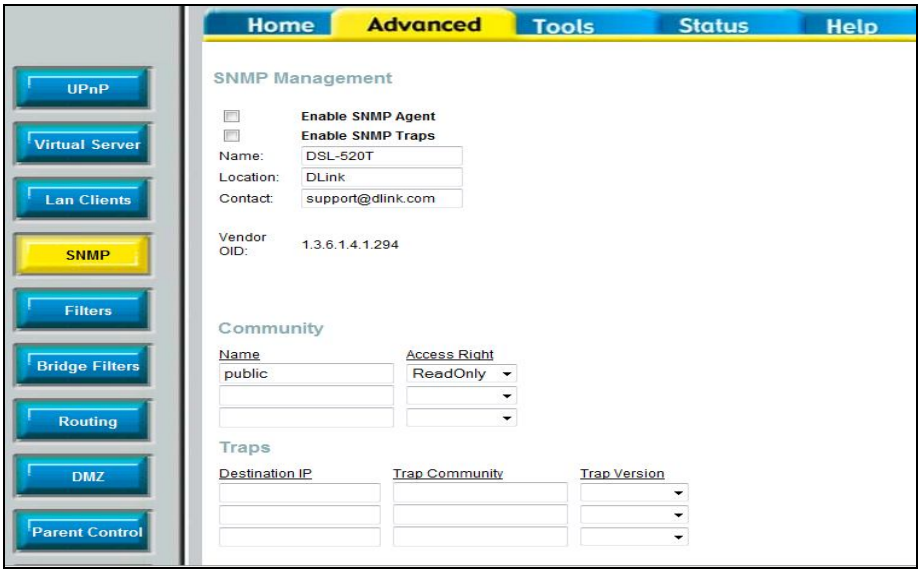
Với bất cứ loại Modem nào, việc cấu hình router cung cấp dữ liệu SNMP theo các bước sau:

Trên Modem ZoomADSL mở trang web cấu hình router, tìm tab Administration -> Management. Trong Management, tìm mục SNMP (hình dưới):

Hình 3.1: Enable SNMP trên Router ADSL ZoomX5, X6

Trong mục SNMP, lựa chọn Enable. Lưu ý xem Modem router của bạn hỗ trợ SNMP version nào và giá trị Community. Trong ví dụ này, Modem router hỗ trợ SNMP V1 và V2, giá trị Get Community là "public"

Trên Modem Dlink-520T (ADSL Router) sau khi login vào cấu hình web, chọn tab Advanced trên menu bên trái chọn mục SNMP.



Hình 3.2: Cài đặt SNMP trên ADSL Dlink-D520T

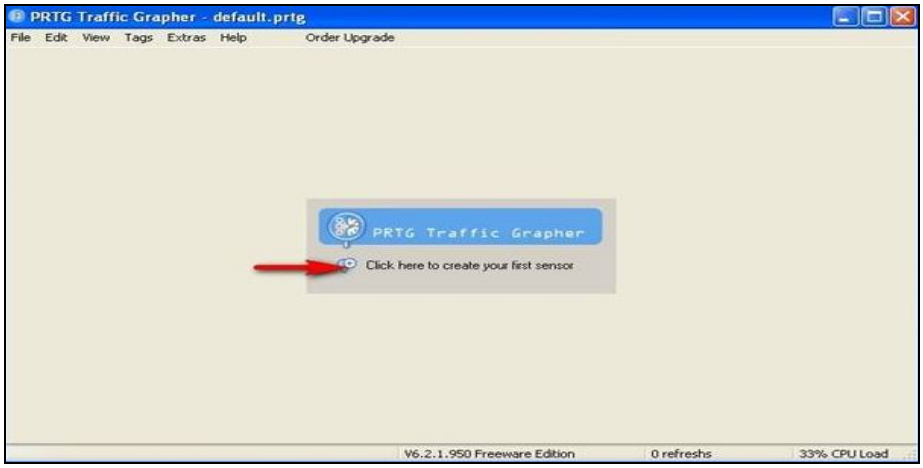
3.2. Phân tích quá trình hoạt động

3.2.1 Cài đặt chương trình

Sử dụng PRTG Traffic Grapher để theo dõi băng thông. Sau khi download PRTG Traffic Grapher, tiến hành cài đặt. Việc cài đặt thực hiện khá dễ dàng. Để chạy chương trình, trên Windows nhắp Start -> Programs -> PRTG Traffic Grapher -> PRTG Traffic Grapher. Xuất hiện hộp thoại Welcome to PRTG Traffic Grapher.

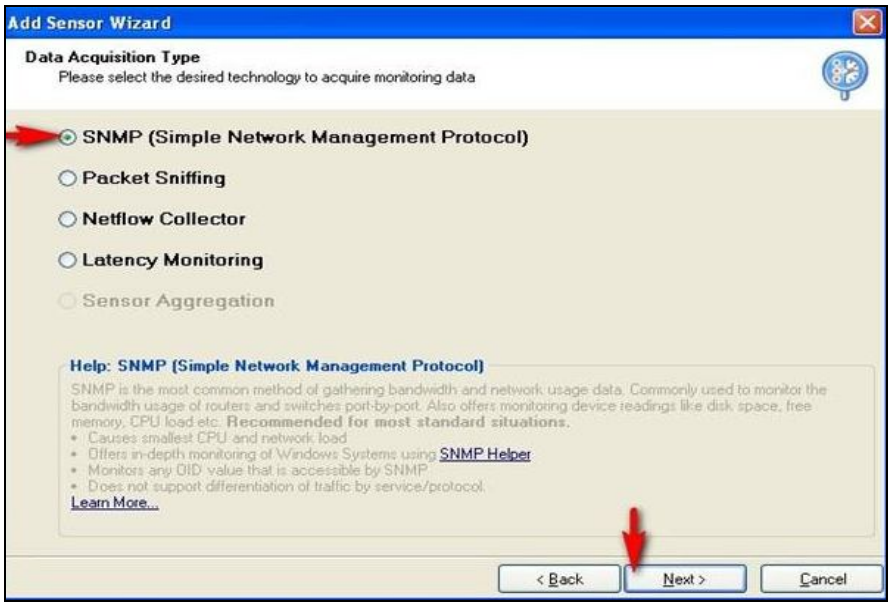
Hình 3.3: Hộp thoại Welcome to PRTG Traffic Grapher

Trên hộp thoại Welcome to PRTG Traffic Grapher, lựa chọn "Use the Freeware Edition", sau đó nhắp Next để tiếp tục. Sẽ xuất hiện giao diện chính của PRTG Traffic Grapher.



Hình 3.4: Giao diện PRTG Traffic Grapher

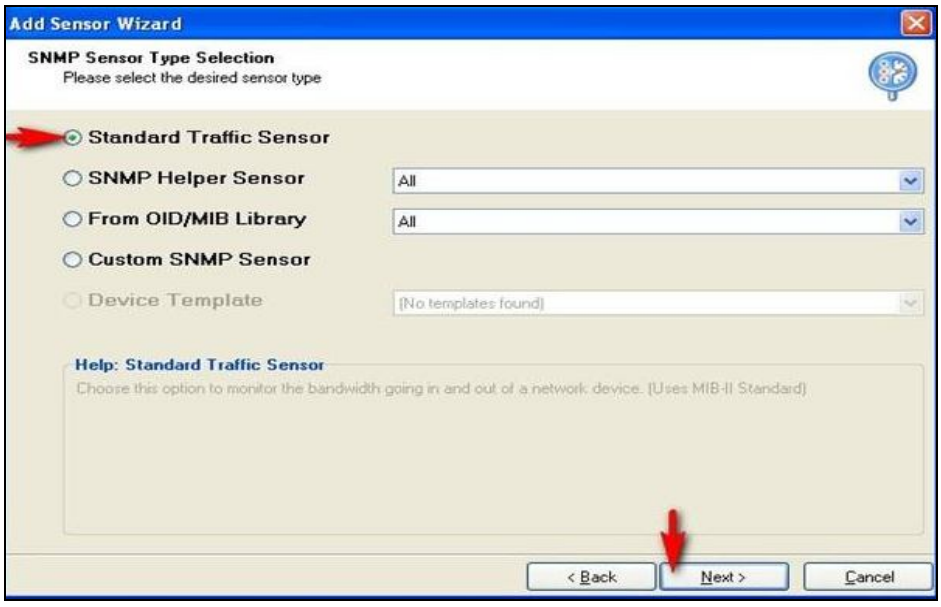
Trên giao diện của PRTG Traffic Grapher, kích chuột vào lựa chọn "Click here to create your first sensor" để tạo các sensor theo dõi. Khi đó PRTG Traffic Grapher sẽ chạy các bước cấu hình để thêm một sensor mới. Dưới đây là các bước cấu hình chính:



Hình 3.5: Chọn giao thức SNMP

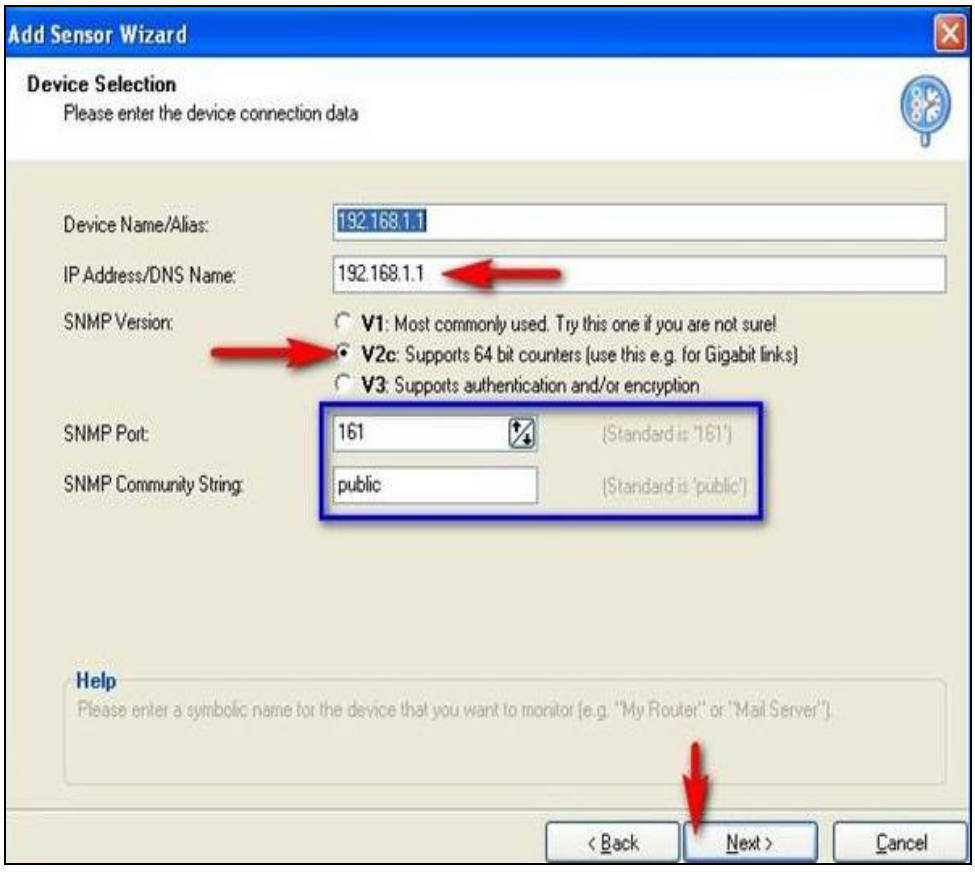
PRTG Traffic Grapher có hỗ trợ các loại sensor: SNMP, Packet Sniffing, NetFlow, Latency. Phiên bản miễn phí chỉ hỗ trợ SNMP và Packet Sniffing. SNMP sensor được sử dụng để theo dõi các traffic IN và OUT của cổng trên Modem router. Packet Sniffing được sử dụng để theo dõi các traffic của card mạng tại máy sử dụng.

Trên hộp thoại Data Acquisition Type, lựa chọn SNMP.



Hình 3.6: Chọn chuẩn Sensor

Trên hộp thoại SNMP Sensor Type Selection, lựa chọn "Standard Traffic Sensor". Các lựa chọn khác tùy thuộc vào thiết bị hỗ trợ SNMP.



Hình 3.7: Lựa chọn IP và version SNMP

Trên hộp thoại Device Selection, xác định các giá trị:

- Device Name/Alias: Nhập tên router do bạn tự quy định. Có thể chọn IP của thiết bị đó cho dễ nhớ.

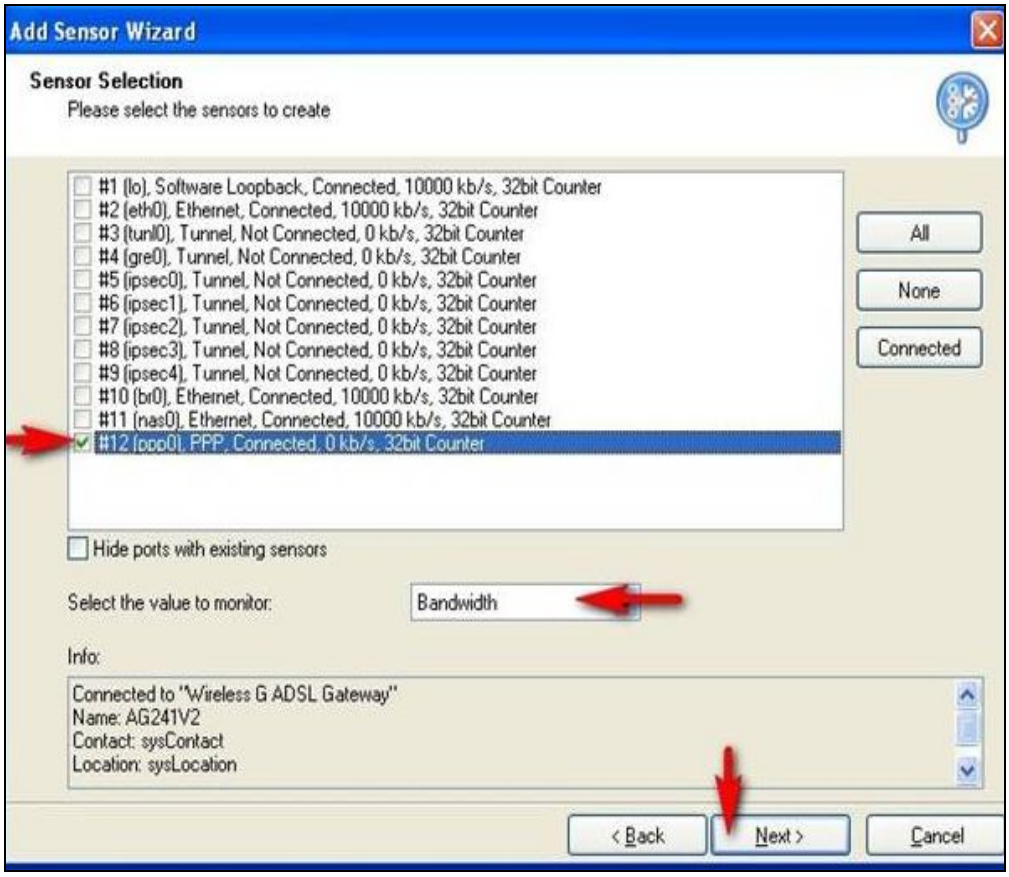
- IP Address/DNS Name: Địa chỉ IP của router.

- SNMP Version: Phiên bản của SNMP trên router hỗ trợ. Trong ví dụ này router hỗ trợ SNMP V1/V2 nên có thể chọn V2c. Có thể chọn lần lượt từng phiên bản để thử.

- SNMP port: Để giá trị ngầm định của cổng SNMP là 161.

- SNMP Community String: Ngầm định là public. Giá trị này có thể kiểm tra ở phần cấu hình SNMP trên router.

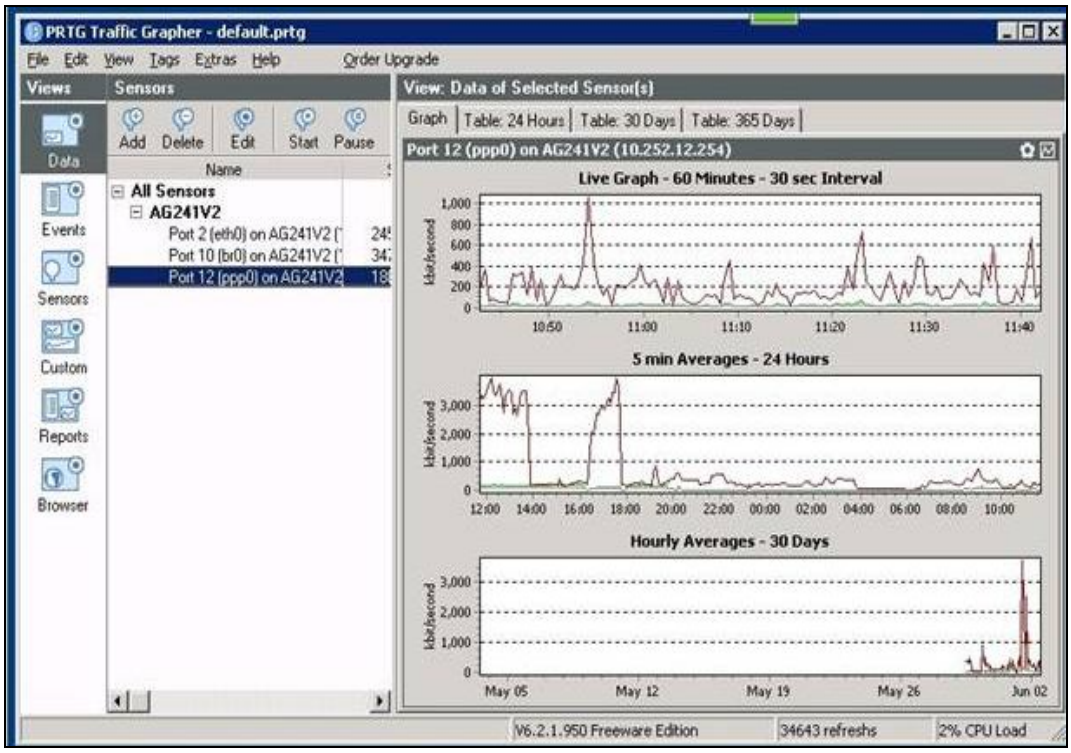
Nếu sử dụng SNMP V3, cần phải xác định SNMP User, Authentication Mode và Password, có thêm lựa chọn là Data Encryption key.



Hình 3.8: Chọn Sensor

Trên hộp thoại Sensor Selection sẽ xuất hiện các cổng mà router đó hỗ trợ. Lựa chọn cổng theo dõi ADSL là ppp0. Lưu ý các ký hiệu: ppp: Point-to-Point Protocol; eth: Ethernet; br: Bridge; wlan: Wireless.

Sau khi lựa chọn cổng theo dõi, xuất hiện giao diện đồ họa theo dõi băng thông của cổng tương ứng.



Hình 3.9: Giao diện Sensor Monitoring

Việc hiển thị được chia thành các loại biểu đồ hỗ trợ người quản trị theo dõi thuận tiện: Live Graph 60 Minutes, Graph 24 Hours, Graph 30 days, Graph 365 days với các mức thời gian và giá trị trung bình khi hiển thị khác nhau.

Lưu ý đường mầu xanh là Bandwith Traffic OUT, đường mầu đỏ nâu là Bandwith Traffic IN. Trên đây là một tính năng của PRTG Traffic Grapher hỗ trợ người quản trị theo dõi băng thông của router ADSL. Các tính năng khác.

3.2.2 Phân tích quá trình hoạt động

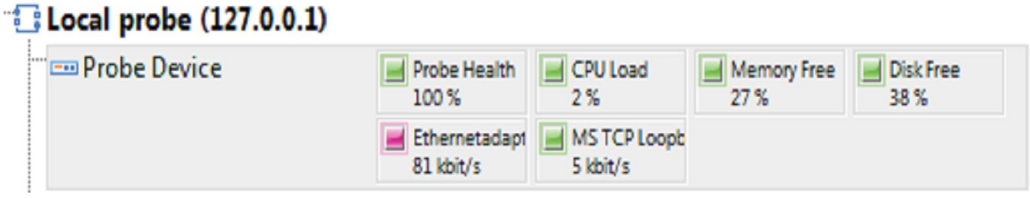
Kiến trúc PRTG Network Monitor bao gồm hai phần chính đó là: PRTG Core Server và PRTG Probe. Vấn đề chính trong quá trình cài đặt PRTG đó là Core Server bao gồm quá trình lưu trữ dữ liệu, web server, các báo cáo và hệ thống lưu trữ. Còn Probe thi hành quá trình giám sát, nó nhận các cấu hình từ Core Server và thực thi quá trình xử lý sau đó báo kết quả về cho Core Server. Một Core Server có thể quản lí không giới hạn các Probe để tăng khả năng giám sát.

Hai phần Core và Probe là hai dịch vụ trong windows chúng chạy bởi hệ điều hành window, không yêu cầu login vào user.

- **Core Server**: là bộ phận quan trọng trong PRTG dùng để xử lí các quá trình

  + Cấu hình quản lí monitor
  + Quản lí và cấu hình kết nối với các Probe
  + Lưu các dòng kết quả của monitor
  + Người quản trị khai báo Mail Server cho quá trình gửi qua Email
  + Lập biểu và báo cáo
  + Quản lí các account
  + Thanh lọc dữ liệu (dữ liệu quá 365 ngày)

- **Probe**: là giao diện PRTG có thể chạy trên một hay nhiều máy tính. Ở quá trình cài đặt được gọi là "Local Probe" tự động được tạo bởi hệ thống. Sau khi nhận được cấu hình từ Core hệ thống tất cả các Probe có thể hoạt động độc lập. Chúng có nhiệm vụ giám sát và thông báo tình trạng hệ thống máy tính.



Hình 3.10: Cấu trúc một Probe

- Trên thẹc tế PRTG Network Monitor thi hành bởi các sensor, mỗi một sensor đại diện cho một thiết bị mạng, có thể là:

+ Một dịch vụ mạng: SMTP, FTP, HTTP…

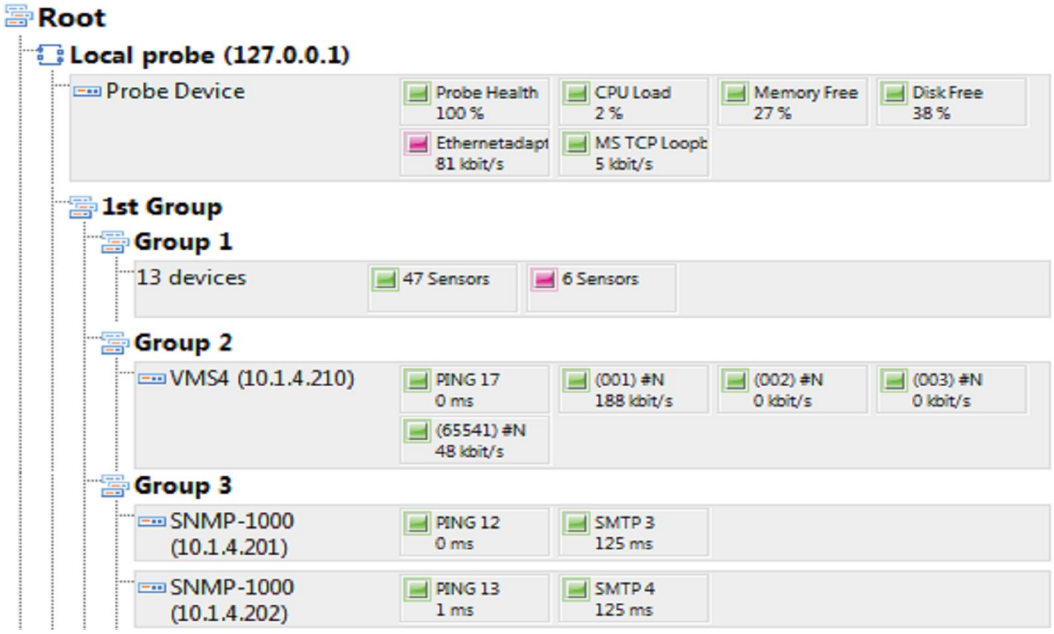
+ Quá trình giao tiếp trên một cổng của Switch

+ Quá trình hoạt động của CPU hay bộ nhớ

+ Quá trình giao tiếp trên card mạng

+ Một thiết bị NetFlow…

- Các Sensor này cho phép User tạo thành các nhóm, mỗi nhóm là tập hợp của một số các thiết bị, mỗi thiết bị lại có tập các Sensor và cuối cùng mỗi sensor có một hoặc nhiều kênh "chanels" (có thể là kênh IN hay OUT)



Hình 3.11: Quá trình gom nhóm các Probe

3.3. Đánh giá hiệu quả mô hình

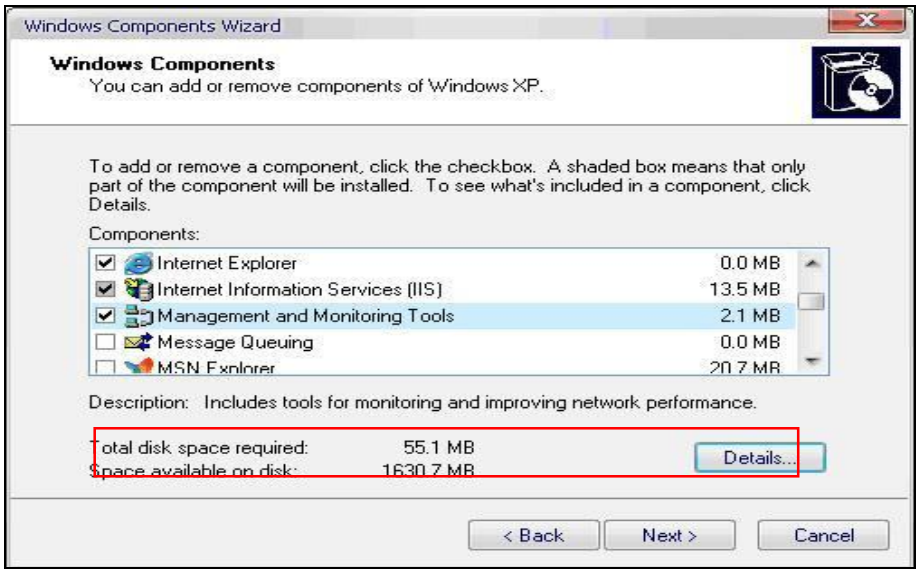
Với thực trạng về nhu cầu an ninh trên Internet như hiện nay thì với những tính năng như trên, có thể nói phần mềm giám sát giao tiếp mạng PRTG sử dụng giao thức SNMP đem lại hiệu quả và độ an ninh cao trong quản trị mạng. Việc sử dụng phần mềm PRTG Traffic Grapher có thể áp dụng cài đặt trong một số mô hình mạng cụ thể.

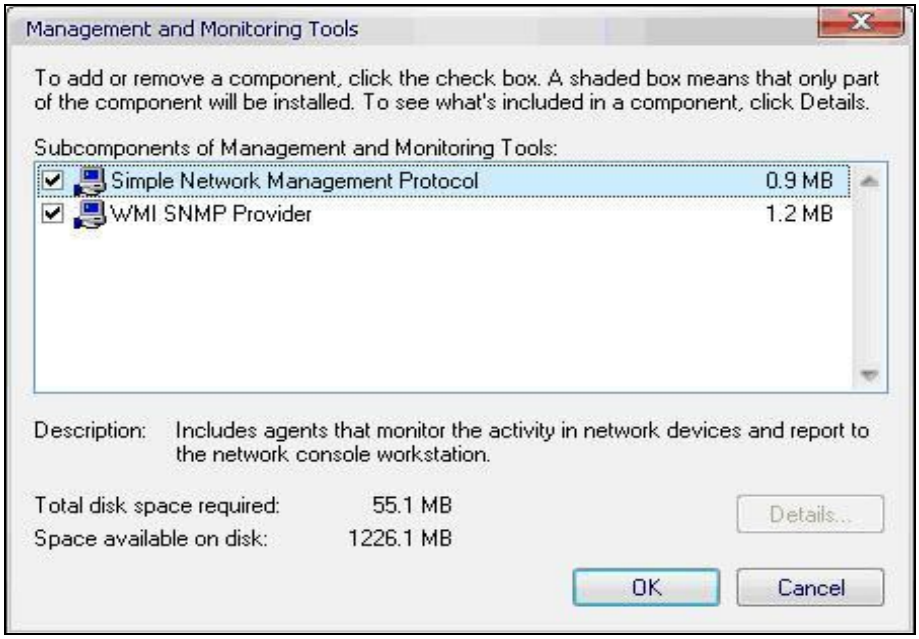
CÀI ĐẶT CẤU HÌNH HỆ THỐNG

1. Cài đặt cấu hình trên Windows

Cài đặt dịch vụ SNMP

Mặc định hệ điều hành Windows không cài dịch vụ hỗ trợ cho giao thức SNMP, để cài thêm ta vào Control Panel, double click vào Add Remove Program → Add Remove Windows Component. Chọn Management and Monitoring Tools.





Khi cài, ta sẽ có thêm 2 dịch vụ hỗ trợ SNMP đó là SNMP Service và SNMP Trap Service:

Cấu hình community trên các dịch vụ của SNMP:

Mở dịch vụ SNMP Service: vào tab Security, tab này cho phép thiết lập community có ý nghĩa giống như mật khẩu giữa thiết bị quản lý và thiết bị cần quản lý. Đối với mỗi community sẽ đi kèm với quyền (rights) khác nhau:

READ-ONLY: Chỉ cho phép đọc, không thiết lập lại thông số được,

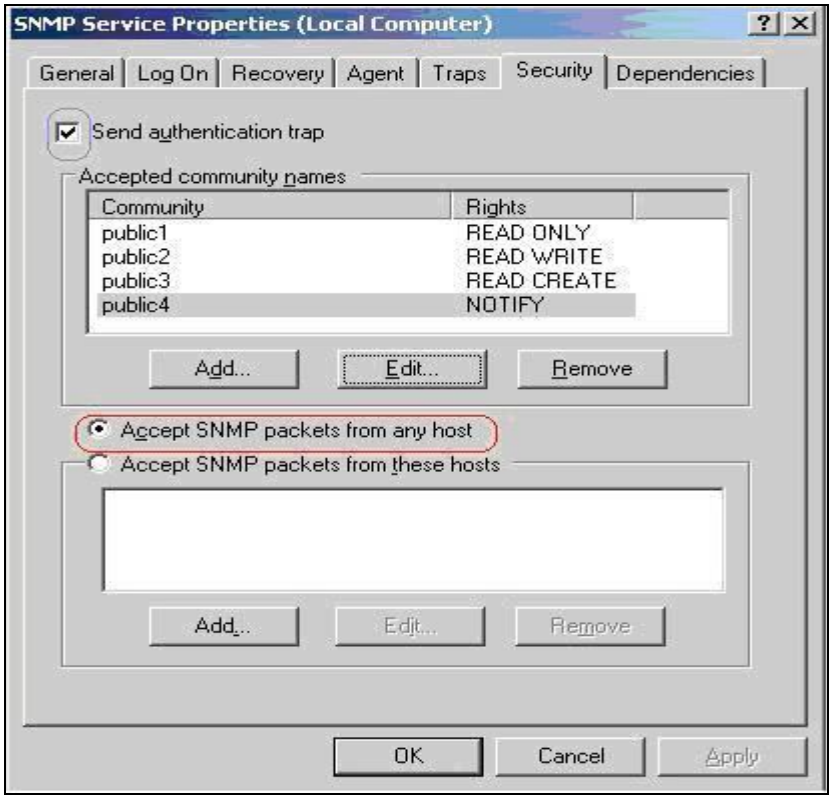
READ-WRITE: Cho phép thiết lập lại thông số.

READ-CREATE: Cho phép tạo ra thông số.

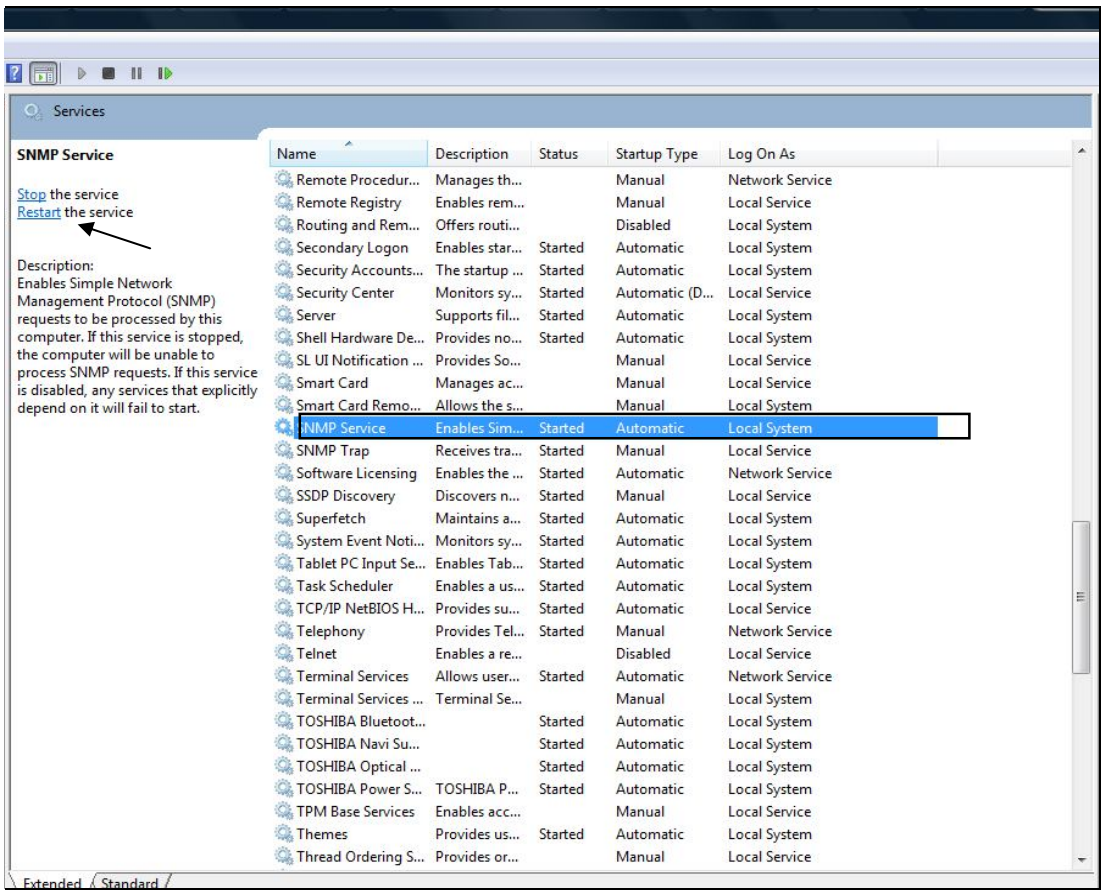
NOTIFY: Dùng cơ chế Trap

NONE: Community không có quyền gì hết.

Đó là thiết lập quyền phía người dùng, bản thân mỗi tham biến cũng có quyền của riêng nó. Ví dụ với tham biến về thời gian UpTime của hệ thống thì ta không set giá trị lại được. Mặt khác, đối với tham biến system.sysContact thì ta có thể đặt lại giá trị được.



Sau khi cấu hình dịch vụ xong cần restart lại dịch vụ: Chọn SNMP Service sau đó kích chuột vào Restart. Làm tương tự đối với SNMP Trap.



2. Cài đặt trên Linux:

Có thể sử dụng gói cài thêm hoặc dùng công cụ Net-SNMP có sẵn trong một số distro Linux.

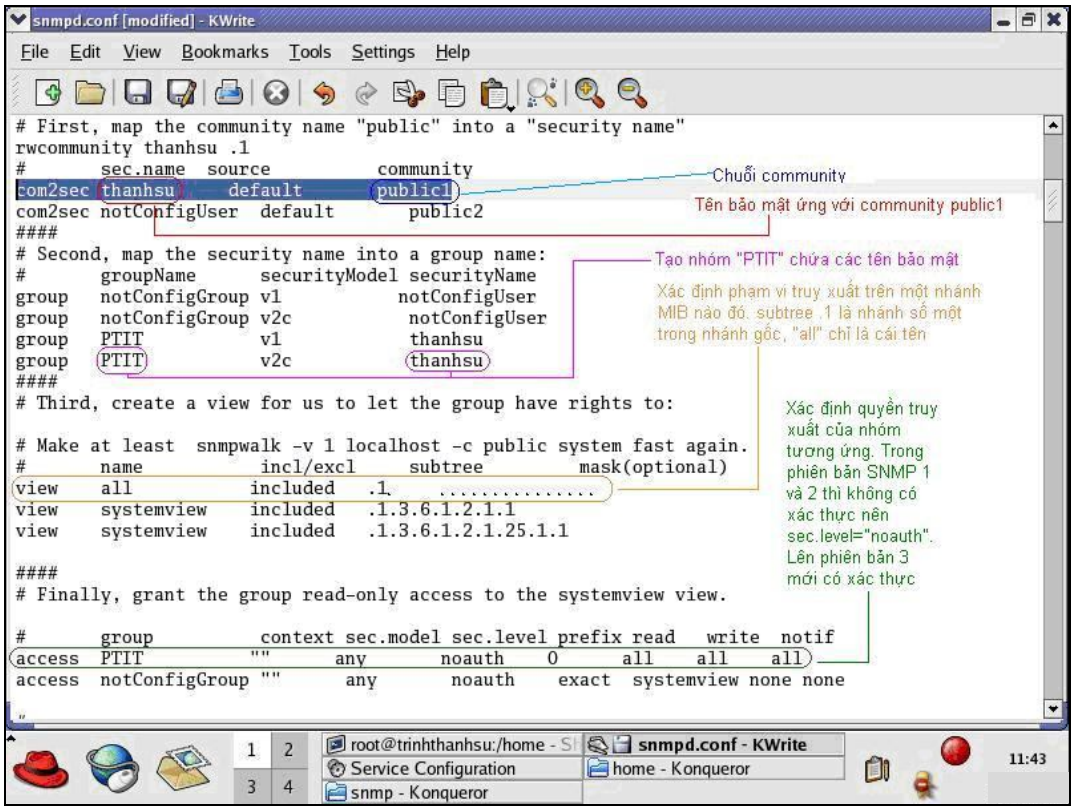
Sử dụng gói cài: giải nén tạo liên kết mềm với Net-SNMP

#cd
#tar-xvzf net-snmp-5.0.2.tar.gz
#ln –s net-snmp-5.0.2 net-snmp
#cd ~/net-snmp
#./configure
…. Biên dịch
#make
…. Cài đặt
#make install
Thiết lập biến môi trường
PATH=$PATH:/usr/local/bin:/usr/loacl/sbin

MIBS=ALL
Export PATH MIBS

Sử dụng dịch vụ SNMP và công cụ Net-SNMP có sẵn của Linux: Mặc định thì dịch vụ SNMP không được cài trên Linux nên phải cài thêm vào. Đối với distro Fedora Core 4 thì cài thêm trong mục System Tools/snmpd. Sau khi cài thì sẽ có thêm hai dịch vụ là snmpd và snmptrapd.

Khác với trong windows, nếu ta muốn thiết lập community cho dịch vụ SNMP thì phải sửa lại file cấu hình /etc/snmp/snmpd.conf như sau:



Sau mỗi lần chỉnh sửa file cấu hình thì gõ lệnh service snmpd restart để khởi động lại dịch cụ SNMP ứng với lần chỉnh sửa mới.

Khi cấu hình xong thì dùng dòng lệnh để quản lý. Phần này giống như trong windows.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết luận

Sau một thời gian thực hiện, em đã hoàn thành luận văn và đạt được một số kết quả nhất định. Trong luận văn này, em đã cố gắng trình bày những kiến thức cơ bản về an ninh trong kiến trúc quản trị mạng SNMP.

Luận văn tập trung chủ yếu vào vấn đề: Tổng quan về quản trị và an ninh thông tin trên Internet, nghiên cứu giải pháp an ninh trong kiến trúc mạng SNMP.

Kết quả đạt được: đã hoàn thành luận văn với các nội dung nêu trên.

Hướng phát triển

Hoàn thành luận văn với kết quả đạt được tương đối theo yêu cầu của đề tài đưa ra. Tuy nhiên trong quá trình thực hiện đề tài, em nhận thấy vẫn còn nhiều vấn đề liên quan cần được tìm hiểu nghiên cứu. Em xin đưa ra một số vấn đề cần tìm hiểu và nghiên cứu phát triển đề tài như sau:

- Dựa vào kết quả nghiên cứu trên có thể xây dựng phần mềm quản trị hệ thống mạng thông qua giao thức SNMP.

- Kết hợp với việc nghiên cứu một số giải pháp an ninh cả về phần cứng và phần mềm khác để có thể xây dựng một hệ thống mạng với an toàn về dữ liệu và an ninh cao.

TÀI LIỆU THAM KHẢO

1. Giáo trình hệ thống mạng máy tính CCNA, Nhà xuất bản LĐXH, 2004.
2. Hướng dẫn thiết lập và quản trị mạng, Nhà xuất bản Thống Kê, 2002.
3. Giáo trình Curicurlum CCNA1 của Cisco System.
4. Internetworking với TCP/IP, Nhà xuất bản Giáo dục, 2001.
5. Computer Security Art And Science, By Matt Bishop, Publisher: Addition Wesley, 2002.
6. Essential SNMP, 2nd Edition, By Douglas Mauro, Kevin Schmidt, Publisher: O'Reilly, Pub Date: September 2005.
7. IETF: RFC2021, RFC1213, RFC1757, RFC2271.
8. http://www.cisco.com/en/US/docs/switches/lan/catalyst5000/catos/4.5/configuration/guide/snmp.html
9. http://net-snmp.sourceforge.net/
10. http://www.paessler.com/download/prtg.