**Remote Monitoring and Network Management Systems**

**Introduction**

Monitoring large switched or routed networks can pose many administration problems. Some network analyzer tools are designed to overcome these problems and are mostly built into management suites such as Microsoft’s SMS or HP OpenView. In most cases the standard SNMP/RMON protocols are employed to capture and analyze the network traffic to detect network faults or problems related to performance, uptime or availability of system components.

WBEM is a DMTF (distributed management task force) initiative and technology that aims to provide a standardized way to access management information in an enterprise environment. Windows Management Instrumentation (WMI) is Microsoft's implementation of WBEM-compatible technologies allowing system problems to be diagnosed and resolved from a central location, thus making networks much easier to manage. Remote control systems such as Telnet, SSH, VNC and Terminal Services are used to assist with network management and many devices now have small web servers built in that allow management via a standard web browser.

**Analyzer Types**

Network analyzers can be standalone or distributed.

Stand-alone analyzers have no difficulty monitoring or capturing data on a simple shared LAN such as Figure 1.



Figure 1 Shared segment analysis

The analyzer is just plugged into a spare port on the hub to see all uni-cast and broadcast traffic on the segment (providing it can operate in promiscuous mode). Capture filters can be set to specify which stations or protocols to monitor or statistics can be gathered for the whole segment.

Problems, however, arise with multi-segment or sub-netted networks containing switches and routers. The same method will not work because they provide point to point connections and the only traffic detectable will be broadcasts or data aimed specifically at the analyzer (i.e. if it is running on the PC you wish to analyze). You could use a portable mini hub as shown in Figure 2 to tap into the point to point segment you wish to monitor. Here the communication between the client and server is being monitored by sharing the server connection with the analyzer, but you still can’t look at the traffic, other than broadcasts, on other segments or full duplex traffic and of course some links don't like being disconnected.

Most managed switches, however, have a feature called 'Port Mirroring' that enables a port to be specified for monitoring so a hub is not required.

*Port Mirroring* can monitor FDX and doesn't require you to disconnect link. It doesn't show all traffic, however, as errors and other packet details such as VLAN membership tags may not be propagated to the analyzer. For this you need a Tap.

*Taps* are special purpose copper or optical splitters for monitoring switch ports that provide complete visibility of all traffic. Taps sit in line with the link and are often left in place, see Appendix for details of a fibre optic tap.





Figurer 2 Monitoring a switched link

*N.B. If the client or server is running a software network analyzer it will see the traffic on the link, but it could impact on link performance.*

Distributed Analyzers such as ‘Observer’ or ‘Sniffer’ (not Wireshark) actually contain two components: an analyzer console to gather and analyze the captured data and remote probes to send data from devices or PCs on other parts of the network.

The combination shown in Figure 3 overcomes the multi-segment analysis problem.



Figure 3 Multi-segment analysis

As the size and complexity of networks increases so does the difficulty of monitoring. For large and enterprise networks sophisticated network management suites are used to monitor and track all aspects of network activity. These packages are based on the industry standard SNMP and RMON protocols.

**SNMP (Simple Network Management Protocol)**

SNMP is a network management standard widely used with TCP/IP networks.

SNMP provides a method of managing network nodes (servers, workstations, routers, bridges, and hubs) from a centrally located Network Management Station (NMS).

This is a large-scale tool for measuring and monitoring network performance.

SNMP uses two network devices:

**Agents** that run on client computers or other networking components such as hubs, routers and switches, to monitor their status and

**Managers** (Network Management Station) that polls clients and summarizes data.

Alerts (called traps) can be set to inform the administrator of any problems.

To perform its management services, SNMP uses a distributed architecture of management systems and agents, as shown in Figure 4.

The centrally located host, which is running network management software, is referred to as an NMS, or an SNMP manager. Managed network nodes are referred to as SNMP agents.

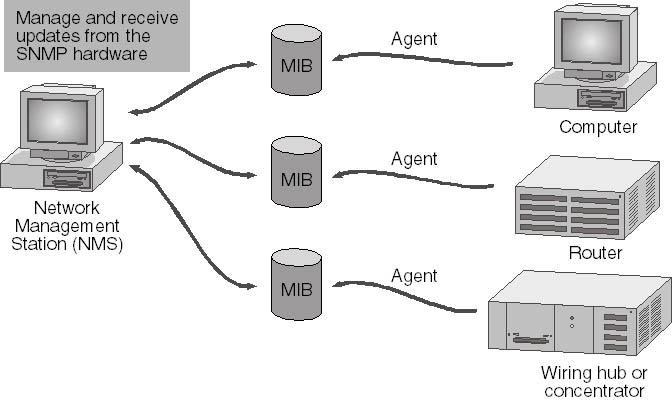


Figure 4 SNMP components

**SNMP Managers**

The NMS does not have to run on the same computer as the SNMP agents.

It can request the information from SNMP agents using SNMP request messages. Information requested may include:

* Network protocol identification and statistics
* Hardware and software configuration data
* Device performance and usage statistics
* Device error and event messages
* Program and application usage statistics

The management system can also send a configuration request to the agent that requests the agent to change a local parameter.

**SNMP Agents**

SNMP Agents provide SNMP Managers with information about activities that occur at the network layer and respond to management system requests for information. The agent service can be configured to determine what statistics are to be tracked and what management systems are authorized to request information, see Figure 5.

In general, agents do not originate messages; they only respond to messages. The exception is an alarm message or alert triggered by a specific event. An alarm message is known as a trap message.

A trap is an event such as a system reboot or illegal device access. Traps and trap messages provide a rudimentary form of security by notifying the management system whenever such an event occurs.

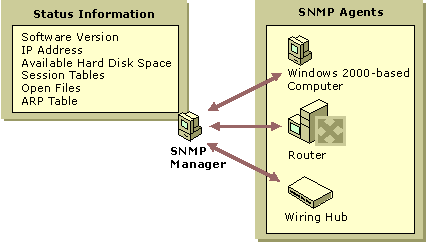


Figure 5 SNMP agents

**Management Information Base (MIB)**

*MIB* is a container of objects (or variables) which represents a particular type of information required by a management system. E.g. one object may represent the number of active sessions on an agent; another can represent the amount of available hard drive space. The information requested from an agent is stored in various MIBs and defines the values for each object. The MIB may contain numbers, text, addresses, object ids and more complex data such as ARP, or routing tables.

Each object in a MIB has a name, syntax, access rights, status and description as specified by ASN.1.

* Name - identifies the object (e.g. SysDesc is the object descriptor and 1.3.6.1.2.1.1.1 is the object identifier)
* Syntax - defines object’s structure (e.g. octet string, integer)
* Access - Objects are Read-only, Read-Write or not accessible
* Status - Mandatory or optional for the particular agent.
* Description - this is a textual description of the object

When requested, the SNMP agent transfers a SNMP message across the network in standard format (specified in RFCs) after being encoded using BER (Basic Encoding Rules) for transmission.

**OIDs (Object Identifiers)**

OIDs are represented by a tree hierarchy – each object has a unique address based on its position in the tree. Figure 6 shows part of the standard hierarchy. The Internet standard management MIB is 1.3.6.1.2.1.2.1 or in text form *iso.org.dod.internet.management.mib*. The root is the ISO trunk value 1. Each branch below further identifies the various branches of the sub-tree. All SNMP objects are members of the sub-tree identified by *iso.org.dod.internet* or 1.3.6.1. Each additional component further defines the exact location of the object- e.g. SysDescr is identified by the OID 1.3.6.1.2.1.1.1.0. The zero at the end indicates that this is a singular ( as opposed to columnar object such as an address table) and has only one instance in the MIB.

*N.B. You can ‘walk’ through the name tree with an SNMP console utility or using a stand alone MIB browser and view the contents of each supported variable in the MIB tree.*

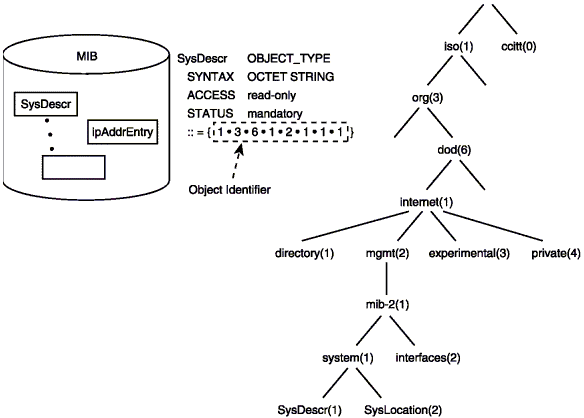


Figure 6 Object name tree

**SNMP Messages**

Both agents and management systems use SNMP messages to inspect and communicate information about managed objects. SNMP messages are sent via the User Datagram Protocol. UDP port 161 is used to listen for SNMP messages and port 162 is used to listen for SNMP traps, see Figure 7.

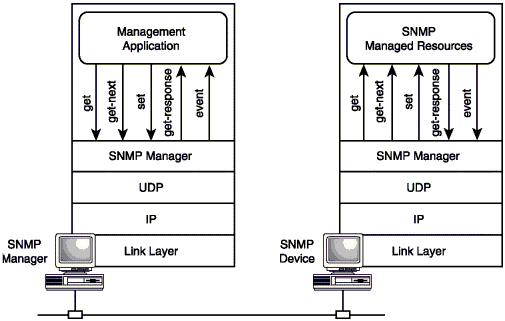


Figure 7 SNMP NMS –AGENT communication

When an NMS sends requests to a network device, the agent on the device retrieves the requested information from the MIBs and sends the requested information back to the initiating NMS. An SNMP agent also sends information when a trap event occurs.

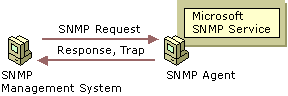


Figure 8 SNMP Messages

In SNMP V.1 the following types of messages are used:

**GET** The basic SNMP request message. Sent by an NMS, it requests information about a single MIB entry on an agent—for example, the amount of free disk space. **GET-NEXT** An extended type of request message that can be used to browse the entire hierarchy of management objects - the agent returns the identity and value of the object that logically follows the previous information that was sent. The GET-NEXT request is useful mostly for dynamic tables, such as an internal IP route table.

**SET** A message that can be used to send and assign an updated MIB value to the agent when write access is permitted.

**TRAP (or NOTIFY)** is an unsolicited message sent by an agent to a management system when the agent detects a certain type of event. E.g. a trap message might be sent when a system restart occurs. The NMS that receives the trap message is referred to as the trap destination.

*N.B. SMNP v.2 introduced GET-BULK A request that the data transferred by the agent be as large as possible within the given restraints of message size. This minimizes the number of protocol exchanges required to retrieve a large amount of management information. SNMP3 is the latest version.*

The interaction between the manager and agents is shown in Figure 9.

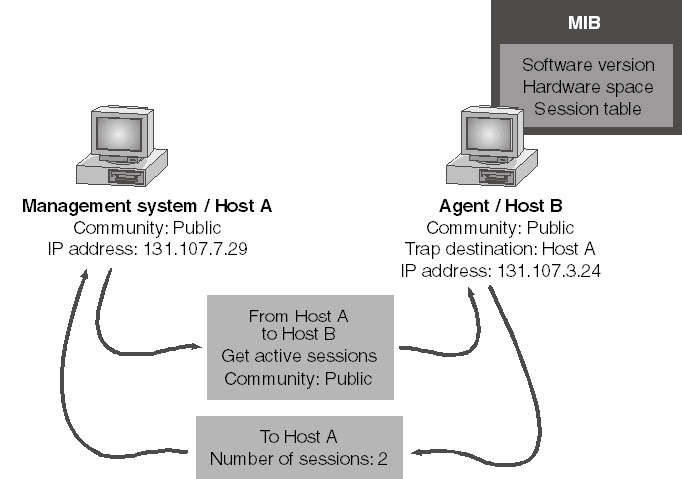


Figure 9 Manager and Agent interaction

The communication process is as follows:

1. NMS forms an SNMP message that contains an information request (GET), the name of the community to which the management system belongs, and the destination of the message—the agent's IP address (131.107.3.24).
2. The SNMP message is sent to the agent.
3. The agent receives the packet and decodes it. The community name (Public) is verified as acceptable.
4. The SNMP service calls the appropriate subagent to retrieve the session information requested from the MIB.
5. The SNMP service takes the session information from the subagent and forms a return SNMP message that contains the number of active sessions and the destination—the management system's IP address (131.107.7.29).
6. The SNMP message is sent to the management system.

**SNMP Communities**

Each SNMP management host and agent belongs to an SNMP community - a collection of hosts grouped together for administrative purposes. Communities are identified by the names you assign to them.

Community names can be used to authenticate SNMP messages and thus provide a rudimentary security scheme for the SNMP service. Although a host can belong to several communities at the same time, an SNMP agent does not accept requests from a management system in a community that is not on its list of acceptable community names.

A community name can be thought of as a password shared by SNMP management consoles and managed computers. By default public communities are read only, private communities are read/write.

In the example illustrated in Figure 10, there are two communities — Public and Public2.

Agent1 can respond to SNMP requests from and can send traps to Manager2 because they are both members of the Public2 community. Agent2, Agent3, and Agent4 can respond to SNMP requests from and can send traps to Manager1 because they are all members of the (default) Public community.

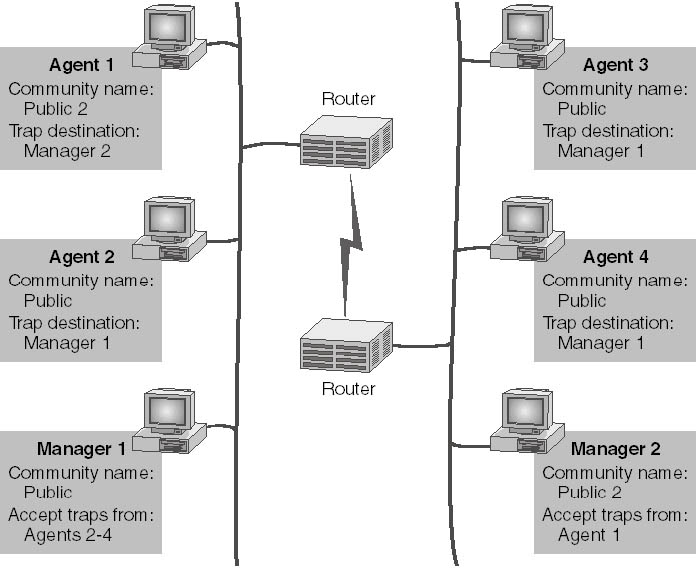


Figure 10 SNMP communities

*N.B. There is no relationship between community names and domain or workgroup names. Community names represent a shared password for groups of network hosts, and they should be selected and changed as you would change any password. Deciding which hosts belong to the same community is generally determined by physical proximity.*

**Windows SNMP service**

The Windows 2000/3 OS provides a SNMP agent installed from the Control Panel Add/Remove Windows Programs application under Management and Monitoring Tools. This contains the Simple Network Management Protocol, which is the SNMP agent which is listed as SNMP Service after it is installed. Once the SNMP service is installed, you can configure the SNMP services through Administrative Tools. In the Services node you select SNMP Service from the details pane, and then select Properties from the Action menu. The SNMP Service Properties dialog box appears, as shown in Figure 11.

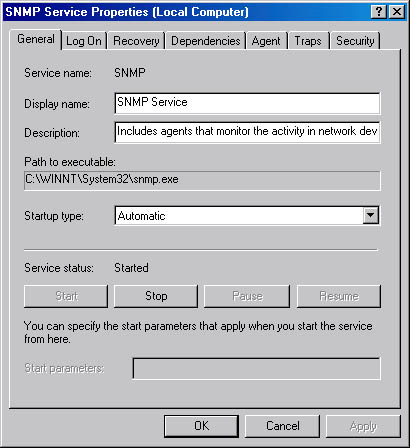


Figure 12 Windows 2000 SNMP properties

*N.B. The SNMP Trap Service is also installed when SNMP is installed.*

**Network Device Agent Configuration and Control**

SNMP managed switches, routers and other network equipment are initially configured using a (out of band) serial port connected to a VT-100 terminal or via a Telnet session (in band). Network equipment vendors also offer management tools to help with agent administration which can be used standalone or integrated into management suites such as HP OpenView. Figure 13 shows the mimic screen for the Allied Telesyn AT-8324 SX switch using SwimView management software.

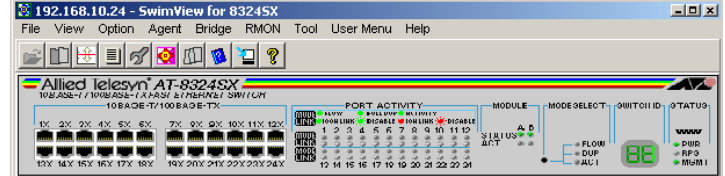


Figure 13 AT8324SX SNMP managed switch

The popular SNMPc, like most of the commercial suites is able to discover and map network components to provide an active multilevel view of all subnets and associated devices.

**Remote Monitoring (RMON) extensions**

Although SNMP can be used to collect network traffic at an interface it is not very efficient in some circumstances. Frequent collection of data can add traffic which could load up the network and requires further processing by the NMS. A better approach is to do some of the processing and data reduction remotely and retrieve data selectively. This is the idea behind the RMON protocol, an extension to SNMP which allows data to be processed at the point of collection. RMON can be implemented in software on an existing device or in hardware called RMON probes. Data is organised in the same manner as with SNMP, described in an RMON MIB and retrieved with SNMP commands. RMON probes appear like super MIBs but provide continuous monitoring and remote data processing.

*N.B. RMON is limited to link level traffic but later versions of the protocol allow for network level information gathering, care must be taken as this requires more processing and can add to network traffic.*

**RMON Groups**

RMON delivers information in nine groups of monitoring elements, but each is optional so that vendors do not need to support all groups within the MIB. These groups are shown Table 1.

RMON II defines another 9 groups for statistics related to upper layer protocols.

*N.B. Alternative protocols are currently being developed for high-speed network traffic monitoring (e.g. InMON traffic server).*

|  |  |  |
| --- | --- | --- |
| **GROUP** | **Function** | **Elements** |
| Statistics | Stats measured by probe | packets/bytes sent dropped, broadcasts/ multicasts, CRC errors, runts, jabbers, collisions |
| History | Trend analysis based on data from stats group | Sample period, number of samples, items sampled |
| Alarm | Configure generation of event if threshold crossed | Type, interval, starting threshold, stop threshold |
| Host | Collects stats based on Host MAC addresses | MAC address, packets, bytes received/transmitted + broadcast, multicast and error packets |
| HostTopN | Collects stats for busiest hosts | Statistics, host(s), sample start and stop periods, rate base, duration. |
| Traffic Matrix | Collects data based on pairs of addresses | Source/destination address – packets, bytes and errors for each pair |
| Filters | Matches packets by filter equation | Bit filter type ( mask or not mask), filter expression, conditional expression |
| Packet Capture | Controls packet capture | Size or buffer for captured packets, full status (alarm), number of captured packets |
| Events | Logs/generates traps | Event type, description, last time event sent. |
| Token Ring | Ring control |  |

Table 1 RMON groups

*N.B. AT- 8324 SX switch used in the netwoks lab supports 4 RMON 1 groups: 1,2,3,9*

**Non SNMP approaches to Network Management**

SNMP is not the only way to retrieve information or monitor remote systems. Apart from SNMP there are a number of remote administration options including third party commercial tools which may be used instead of or as a compliment to SNMP.

A number of devices now have small HTTP servers built in that allow remote configuration and management of access points, routers etc. using a GUI e.g. Figure 14 shows the AT- 8324SX managed switch web agent which allows all aspects of the switch to be controlled and monitored.

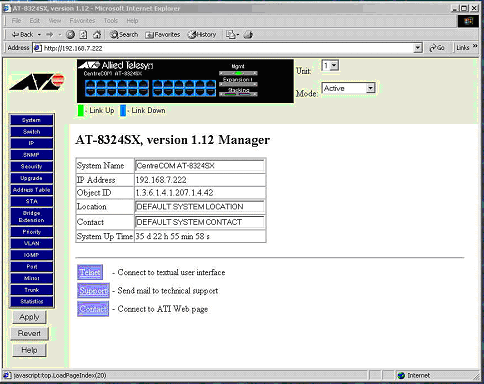


Figure 14 AT-8324SX Web management console

Many network devices also have Telnet (or the more secure SSH) server capabilities to allow connection and management over the network.

Figure 15 shows the Telnet screen for the AT AR310 ISDN router. This allows most aspects of the device to be configured and monitored locally or remotely including SNMP setup.

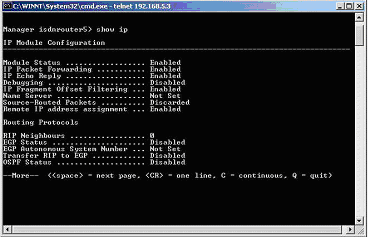
.

Figure 15 Telnet screen for AR-310 router

**Remote Control options**

If remote access is the only consideration then remote control applications could be used. VNC is a popular choice for UNIX and is also available for Windows, but doesn’t allow multi user access for Windows.

The administrative mode of Terminal Services is also an attractive management option for Windows 2000/3 Server, see Figure 16. All aspects of server administration can be carried out over the network using Terminal Services just as if the administrator were sat at the server console. The XP remote desktop client can be used to remotely manage other XP Pro PCs or Terminal Servers.

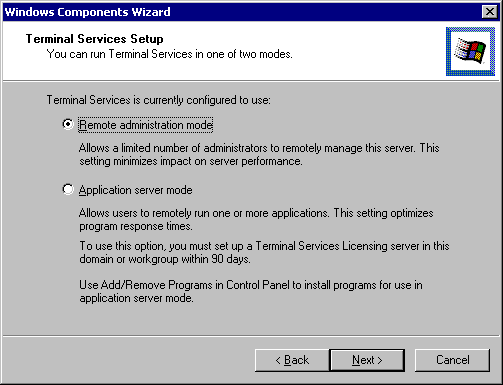


Figure 16 Terminal Services setup

**Web-Based Enterprise Management (WBEM) and WMI**

WEBM is an industry initiative to develop a standardized, non-proprietary means for accessing and sharing management information in an enterprise network. It is intended to solve the problem of collecting end-to-end management and diagnostic data in enterprise networks that may include hardware from multiple vendors, numerous protocols and operating systems, and a legion of distributed applications. WBEM provides this common model, shown in Figure 17a, and data source, and can be extended to work with existing network components, tools, and protocols.

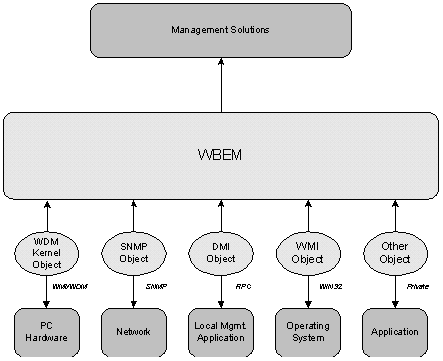
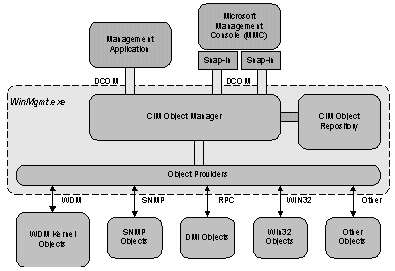


Figure 17aWEBM model Figure17b WMI model

WMI (Windows Management Instrumentation) is Microsoft's implementation of WBEM. It provides a fully integrated operating system support for uniform system and application management based on the CIM by providing a descriptive model of the operational, configuration, and status aspects of Windows based systems, see Figure 17b.This model can be used to instrument management applications using a standard mechanism for storing object definitions (a CIM-compliant object repository), a standard protocol for obtaining and disseminating management data (COM/DCOM; other protocols are also possible), and one or more Win32 dynamic link libraries (DLLs) that function as WMI data providers.

**Network Discovery and Mapping**

Management tools such as NetworkView and SNMPc can scan a network using SNMP, WMI and ICMP to show the devices on a network map such as the one shown in Figure 18. A useful feature to aid network management.

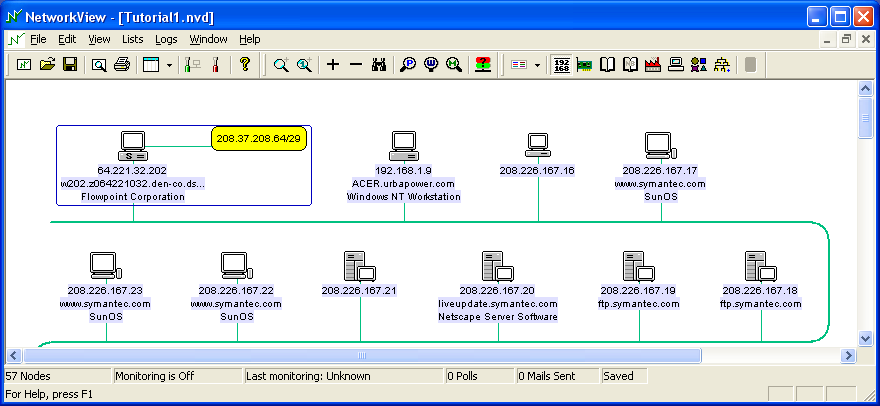


Figure 18 discovered map of a network

Appendix

Enabling *Ping Scan Subnets* will force SNMPc to check every IP address on a subnet where SNMPc discovers a device. This is a useful option if you want SNMPc to discover all the devices on a network. If you enable this option you should also set the *Filter* option so that the subnets being ‘scanned’ can be limited. It is not recommended that you enable Ping Scan Subnets if you are using the periodic Auto discovery.

As default the auto-discovery will restart every hour. This is controlled by the *Auto Restart Time* (hours) setting. In the above example this has been changed to 12. Auto-discovery will now run twice a day. To disable the auto-discovery from restarting set the Auto-Restart time to 0

When SNMPc discovers a new icon it will add it to the map. You can control how the icon is added using the *Layout* option. The default is Top Level/Complete. This will add the device to the existing map and rearrange any existing icons for the ‘best fit’. If you have manually moved icons on your map for aesthetic reasons you should select Top Level/Incremental. This will add the icon to the map but will not move any existing icons.

By selecting Discovered Objects as the layout option new devices are added under a new submap named ‘Discovered Objects’. This is a useful way of identifying newly discovered devices. You can subsequently move the new icons from Discovered Objects submap to the main map.



