The Simple Times™

THE BI-MONTHLY NEWSLETTER OF SNMP TECHNOLOGY, COMMENT, AND EVENTS℠
VOLUME 1, NUMBER 5 NOVEMBER/DECEMBER, 1992

The Simple Times is an openly-available publication devoted to the promotion of the Simple Network Management Protocol (SNMP). In each issue, ***The Simple Times*** presents: a refereed technical article, an industry comment, and several featured columns. In addition, some issues include brief announcements, summaries of recent publications, and an activities calendar. For information on submissions, see page 20.

In this Issue:

Technology and Commentary
Technical Article 1
Industry Comment 5

Featured Columns
Applications and Directions 5
Ask Dr. SNMP 7
Security and Protocols 8
Standards 10
Working Group Synopses 13

Miscellany
SNMP Usage Survey 19
Recent Publications 19

Publication Information **20**

The Simple Times is openly-available. You are free to copy, distribute, or cite its contents. However, any use must credit both the contributor and ***The Simple Times***. (Note that any trademarks appearing herein are the property of their respective owners.) Further, this publication is distributed on an "as is" basis, without warranty. Neither the publisher nor any contributor shall have any liability to any person or entity with respect to any liability, loss, or damage caused or alleged to be caused, directly or indirectly by the information contained in ***The Simple Times***.

The Simple Times is available via both electronic-mail and hard-copy. For information on subscriptions, see page 20.

Technical Article

Allan Leinwand, Cisco Systems, Inc.

In this issue: *Accomplishing Performance Management with SNMP*

Performance management is the process of measuring the performance of all elements which comprise a data network. This involves procedures to find the current utilization of network links and segments, identifying areas of possible congestion, isolating high error rates, and examining network traffic patterns. Each of these areas can aid the network manager to ensure that the network performs to the user's expectations. Performance management techniques can help you work on current network problems concerning slow response time and identify long-term trends which need attention.

Network managers can use SNMP to retrieve information found in MIB-II (RFC 1213) and the RMON MIB (RFC 1271). This information, if applied appropriately, can be used to accomplish many aspects of performance management. The MIB-II and the RMON MIB documents specify the data available from network devices which is found in individual pieces called objects. An object may be nearly any piece of information, ranging from a text description of the device to a number denoting the total number of packets sent on a single interface. MIB-II describes the objects available from any device which runs the Internet suite of protocols, while the RMON MIB defines objects available from remote network monitoring devices. The RMON MIB is especially useful in providing information about a segment which may have devices which do not "speak" SNMP. These two documents form the basis of standard information available from many internetworks.

Software developers, working in the arena of network management, and network managers need to understand how to apply the objects of MIB-II and the RMON MIB toward accomplishing performance management. Performance management information often requires looking at a certain statistic over a period of time. This period of time can vary from a few seconds to a month or more. For example, you may want to look at the errors of a serial link every few seconds. Yet in other situations, you may need to examine the trend of overall utilization

on a token ring segment for the past month. Both of these cases require you to calculate a *delta* to determine the overall change in the statistics. Since calculating deltas will be prevalent in this article, we will use the syntax:

```
delta(X,t1, X,t0)
```

to denote the change between the statistic X at time t0 and time t1.

Calculating Link/Segment Utilization

One possible performance management application is to calculate the utilization of a link. Calculating the link utilization can be used to isolate current performance problems or help avoid congestion through long term capacity planning. The examination of the utilization on network links is an important step toward accomplishing performance management.

The link utilization that will adversely affect network performance will depend on many factors including the underlying data link protocol, the retransmission algorithm being used by hosts, and the applications using the link. Because of these variables, many organizations will have different percentages of link utilization which will result in the users experiencing degraded performance. This poor performance is often characterized by slow response time.

Using objects from the Interfaces group of MIB-II one can find the utilization percentage for a single device on a broadcast media (such as an Ethernet segment). Using the same objects on a full-duplex point-to-point link (i.e., HDLC, PPP, and so forth), can provide enough information to calculate the utilization of the media.

The objects `ifInOctets` and `ifOutOctets` give the total number of bytes received and sent on an interface. Examining the deltas for these numbers and dividing by the bandwidth results in utilization percentage. The bandwidth in kilobits per second of an interface is found in the object `ifSpeed`. As an example, the utilization on an Ethernet interface may be found using this formula:

```
utilization =
    ( 8 * (  delta(ifInOctets,t1,
                   ifInOctets,t0)
           + delta(ifOutOctets,t1,
                   ifOutOctets,t0))
        / (t1 - t0))
    / ifSpeed
```

You need to multiply the delta of total bytes received and sent by 8 to convert the units of `ifInOctets` and `ifOutOctets` (bytes) to the units of `ifSpeed` (bits). Note that this formula will calculate the interface utilization, not the utilization of the entire media.

On full-duplex point to point media, you will need to change the formula to only use the greater of input or output bytes. If you do not do this, you could potentially calculate a 200% utilization (full bandwidth in both directions simultaneously)! The following formula works in these situations:

```
utilization =
    ( 8 * max(delta(ifInOctets,t1,
                    ifInOctets,t0),
              delta(ifOutOctets,t1,
                    ifOutOctets,t0))
        / (t1 - t0))
    / ifSpeed
```

By looking at the object `ifType`, you can determine which of the above utilization formulas applies to the current interface. For example, the following `ifType` numbers correspond to potential full-duplex interfaces: 2 (`regular1822`), 3 (`hdh1822`), 4 (`ddn-x25`), 5 (`rfc877-x25`), 16 (`lapb`), 17 (`sdlc`), 18 (`ds1`), 19 (`e1`), 20 (`basicISDN`), 21 (`primaryISDN`), 22 (`proprietaryPointToPointSerial`), 23 (`ppp`), 28 (`slip`), 30 (`ds3`), 31 (`smds`), and, 32 (`frame-relay`).

For certain media types such as X.25 or Frame Relay, it might make sense to look at the utilization at both ends of the virtual circuit where the local access devices attach to the backbone network. You can then use this data to determine if the utilization of all the access links to the network will congest the backbone.

The objects from MIB-II can enable you to find the utilization of an interface on a multicast media but not the utilization of the entire segment. Determining segment utilization can be accomplished by using the object `etherStatsOctets` found in the Statistics group of the RMON MIB. The `etherStatsOctets` object gives the total number of octets transmitted on the attached Ethernet segment. Computing the delta of this object, multiplying by 8 (to convert from bytes to bits) and dividing by the bandwidth of the segment (found in `ifSpeed`) allows you to determine the utilization of an Ethernet segment.

For long-term performance management of an Ethernet segment, the History group of the RMON MIB contains an object, `etherHistoryUtilization`, which can be used. (The History group stores information that was gathered by the Statistics group for later analysis.) The `etherHistoryUtilization` object gives a best estimate of the mean physical layer network utilization for the interval that the statistics are kept.

Managing Congestion

Congestion is the point at which the overall throughput on a link reaches zero because the bandwidth does not

have the capacity to transmit data at a rate which does not result in error or retransmission. This may result in transport protocols which require acknowledgment (i.e., TCP) resending data. If this occurs often, it can severely degrade network throughput and response time. Congestion can also be seen as the point in which the delay to get data across a network becomes infinite. By looking at the utilization of the links on the network you may be able to avoid congestion by increasing bandwidth or redesigning traffic flow. In addition to looking at utilization, MIB-II provides some objects which may help you determine if congestion is about to become a problem on a link.

The objects `ifInDiscards` and `ifOutDiscards` tell you the number of packets which were chosen to be discarded by a system even though no errors had been detected on input or output. One potential reason for the system to discard packets is because of a lack of buffer space. Buffer space is the area of memory used to hold packets while they are being received. The buffers on a system could be filled because the interface they are routed out does not have the bandwidth to permit the sending of packets at the same rate as the receipt of packets. Another reason may be that the system is busy executing other processes and does not have adequate resources to receive the data being sent. Regardless of the cause, an increase in the number of input or output discards can result in the retransmission of packets. If continual retransmission occurs without any throughput, this may cause congestion and poor network performance.

When the system sends a packet, it first places the packet on the output queue of the proper interface. The packet is queued until the interface can transmit it, which may not occur immediately if the link is currently occupied. The object `ifOutQLen` gives you the length of this queue for each interface, counted in packets. If packets are held in this queue long enough (or held in multiple buffers and queues en-route) then the source system may retransmit the packet. This situation could also occur if there were errors on the link preventing the transmission of error free packets. If the object `ifOutQLen` increases, this could give you a clue as to an impending performance problem.

Because retransmissions can be a hint toward possible congestion, the TCP group of MIB-II tells you the retransmit algorithm being used in the system by the object `tcpRtoAlgorithm`. Certain retransmission methods can help avoid congestion, such as the use of Van Jacobson's algorithm. Further, the number of TCP segments retransmitted by the system is found in the object `tcpRetransSegs`. If this number increases dramatically you could expect to see a degradation in throughput for applications using TCP.

Calculating Error Rates and Percentages

Link errors can affect network performance in many of the same ways as high link utilization. Errors on a network link can cause congestion, low throughput, and slow response time. Looking at the errors on a network link in real-time can help isolate current network problems. Examining the trend in errors over a longer period of time can help you correct the errors before they adversely affect network performance.

The objects `ifInErrors` and `ifOutErrors` in MIB-II give the number of input and output errors for a network interface. Calculating the delta of these objects over time shows the rate of errors on an interface:

```
input-error-rate  =
      delta(ifInErrors,t1, ifInErrors,t0)
    / (t1 - t0)

output-error-rate =
      delta(ifOutErrors,t1, ifOutErrors,t0)
    / (t1 - t0)
```

In many situations it is more relevant to examine the percentage of input and output errors in relation to the amount of total traffic. For example, observing 50 errors per second on an Ethernet segment might seem to be a problem until you examine that the Ethernet segment is passing 3000 packets per second, resulting in less than 2% errors. The objects `ifInUcastPkts` and `ifInNUcastPkts` give you the total input unicast and non-unicast packets on an interface. Likewise, `ifOutUcastPkts` and `ifOutNUcastPkts` do the same for output packets. The input and output rate of packets on an interface are calculated:

```
input-packet-rate  =
      delta(ifInUcastPkts,t1,
            ifInNUcastPkts,t0)
    / (t1 - t0)

output-packet-rate =
      delta(ifOutUcastPkts,t1,
            ifOutNUcastPkts,t0)
    / (t1 - t0)
```

Thus, calculating the percentage of input and output errors for an interface follows these formulas:

```
input-error-percent  =
    input-error-rate / input-packet-rate

output-error-percent =
    output-error-rate / output-packet-rate
```

Often examining the input and output error percent separately can help you isolate a network problem. Input errors possibly indicate problems with the data being

received (such as frames which are too large or too small) or transmission clocking issues. In some cases, output errors can be the result of problems seen on the physical network media, such as a sync loss on a serial link, or with the source system.

The RMON MIB has objects in the Host group which can help you determine the output error rate for a host using the objects `hostOutPkts` and `hostOutErrors`. This is particularly useful if you are trying to accomplish performance management on a segment which has hosts that do not have an SNMP agent and that does have a device that supports the RMON MIB.

In addition to the interfaces group, MIB-II provides objects to help find errors dealing with the IP, ICMP, TCP, UDP, and EGP protocols. Information regarding each protocol is contained within a group. It is important to appreciate that errors do not propagate between groups. For example, if you observe 5 errors per second on a serial interface and at the same time record 5 UDP errors per second, this is merely a coincidence. Yet, this information can still be useful as hints to help isolate a problem. While also helping to isolate faults, errors on the system can directly affect the performance of the network.

For IP, the objects `ipInHdrErrors` and `ipInAddrErrors` in the IP group can help isolate if the errors are due to datagrams which were discarded because of errors in their datagram headers. IP input errors can also occur if the destination address was not valid for the system on which the agent resides. The rate that these objects change, as compared to the input and output error rate, can help you isolate IP errors. To examine this further, you may wish to find out the percentage of IP datagrams received which were errors. Using the same methods as described above for calculating the input and output error percent, you can use the ipInDelivers object to find the total IP datagrams received and then find the percentage of IP input errors:

```
ip-error-rate =
    ( (  delta(ipInHdrErrors,t1,
               ipInHdrErrors,t0)
       + delta(ipInAddrErrors,t1,
               ipInAddrErrors,t0))
    / (t1 - t0))

ip-input-rate =
      delta(ipInDelivers,t1,
            ipInDelivers,t0)
    / (t1 - t0)

ip-input-error-percent =
    ip-error-rate / ip-input-rate
```

The ICMP group has the objects `icmpInErrors` and `icmpOutErrors` which give the total number of ICMP errors received and sent by the system. The delta for these objects as compared to the input and output error rates can tell you how many error datagrams result from ICMP. The objects `icmpInMsgs` and `icmpOutMsgs` give the total input and output ICMP messages. In the manner described above for IP datagrams, you can use these to compute the ICMP input and output error percentages.

The objects `tcpInErrs` and `tcpOutErrs` of the TCP group give you the total number of errors in the receipt and transmission of TCP segments. The objects `tcpInSegs` and `tcpOutSegs` count the total input and output TCP segments. These objects provide the information necessary to find out the percentage of TCP errors.

Similarly, the UDP group from MIB-II has the objects `udpInDatagrams`, `udpOutDatagrams`, and `udpInErrors`. These objects give you the total number of input and output UDP datagrams and input errors. With these objects you can find the percentage of UDP packets received which were in error.

The objects `egpInErrors` and `egpOutErrors` found in the EGP group tell the number of received and sent EGP errors. The objects `egpInMsgs` and `egpOutMsgs` give the total number of EGP messages received and sent. Given this information, you can find the percentage of EGP messages which resulted in errors. There is also a table of information, the `egpNeighTable`, which can help you isolate which EGP neighbor is causing the errors. While EGP does not directly affect the performance of the network (it just provides network reachability information), processing excessive EGP error messages could hamper the performance of a system.

The SNMP group also provides objects which give you the necessary information to calculate the percentage of errors generated by sending or receiving SNMP packets. The SNMP group objects `snmpInPkts` and `snmpOutPkts` give the total input and output SNMP packets. The various SNMP errors are then broken down further in separate objects. With these objects you can calculate the rate of errors in SNMP input and output packets.

MIB-II provides a way to examine errors on a system on a network interface. However, it is often advantageous to look at the errors on the entire segment of a network to help accomplish performance management. Errors on an Ethernet segment can result in poor network performance, especially if the errors result in excessive collisions and retransmissions.

To this end, the RMON MIB provides objects in the Statistics group concerning errors on an Ethernet segment. The objects

```
etherStatsCRCAlignErrors
etherStatsUndersizePkts
etherStatsOversizePkts
```

```
etherStatsFragments
etherStatsJabbers
etherStatsCollision
```

each provide important statistics on the segment. These objects count the network checksum (CRC) errors, packet size errors, and collisions. You can calculate the percentage of errors on a segment by computing the delta for any of these objects and dividing by the delta of the total packets on the segment, given by the `etherStatsPkts` object.

Objects which can store these values for later analysis are found in the History group of the RMON MIB. Looking at trends in errors or collisions can help you plan for the future needs of the network or avoid an increase in errors which will affect performance.

Determining Traffic Patterns

The pattern of traffic on a data network can also affect performance. Important factors to examine are the type of traffic and the flow of traffic between hosts.

The type of traffic on a segment can be broken into two categories: non-broadcast and broadcast. Non-broadcast traffic is destined for a single host on the segment while broadcast traffic is received by all hosts on the segment. A large amount of broadcast traffic can affect performance because it requires processing power by all attached hosts to process the traffic. Also, in a transparent or source-route bridged environment, broadcasts are forwarded to all segments which may consume bandwidth.

You can count the broadcast traffic rate on an interface using the `ifInNUcastPkts` and `ifOutNUcastPkts` objects from MIB-II:

```
broadcast-rate =
      delta(ifInNUcastPkts,t1,
            ifOutNUcastPkts,t0)
    / (t1 - t0)
```

Likewise, you can count the amount of non-broadcast traffic on an interface using this formula:

```
traffic-rate =
      delta(ifInUcastPkts,t1,
            ifOutUcastPkts,t0)
    / (t1 - t0)
```

The RMON MIB's Host group gives statistics about the type of traffic each host on the segment is sending. You can use the objects `hostOutBroadcastPkts` and `hostOutMulticastPkts` to compute the type of traffic sent by each host on the segment.

To examine the flow of traffic on a segment, you can use the `matrix` table found in the Matrix group of the RMON MIB. This table gives source and destination address pairs along with packets (`matrixSDPkts`), bytes (`matrixSDOctets`), and errors (`matrixSDErrors`) sent between them. These counters will be useful to determine which devices control the flow of traffic (i.e., routers) and those that dominate network activity (e.g., servers).

Examining the rate at which pairs of hosts communicate can help you determine how to segment a network with a device such as a bridge or router. For example, after observing the rate of traffic between two hosts you may decide to put them on the same interface of a router. If it is possible to restructure the network based upon the traffic between the most active hosts (perhaps arranging to have as few devices as possible between them) this can improve the network response time and performance.

Summary

The objects found in MIB-II and the RMON MIB can help you accomplish performance management through the use of SNMP. MIB-II is supported on all standard SNMP-speaking devices, while the RMON MIB gives statistics about all devices on a local segment. To help keep the network performing to the satisfaction of the user involves calculating link and segment utilization, managing congestion, calculating error rates and percentages, and determining network traffic patterns. Each of these tasks can be done by applying MIB-II or RMON MIB objects appropriately.

Industry Comment

Marshall T. Rose

This issue is right at the page count limit, so "no comment". By the way, there are now over 1650 electronic subscribers (including several re-distribution lists).

Applications and Directions

Steven L. Waldbusser

In this issue: *How RMON stands to replace the traditional protocol analyzer*

The Remote Network Monitoring MIB (the RMON MIB) was published as an RFC one year ago and has been widely implemented since then. Now that many products are available that implement the RMON standard, it is easier to speculate with some confidence the impact that RMON will have on the network management industry. One of the most profound impacts that the RMON MIB will have is to displace the traditional protocol analyzer

in both large and small network environments. This will happen because an RMON solution can be more cost-effective, can result in better applications, and can be more easily integrated with other necessary network management tools.

More cost-effective

A simple network analysis system based on the RMON MIB consists of several RMON probes and one RMON management station. The probes gather network packets and statistics by reading all data on the network, much like a protocol analyzer. This data is formatted according to the RMON standard and made available to SNMP requests from the RMON management station. The RMON management station formats the data and presents it to the network manager in the most effective way possible (often using a graphical user interface). Because RMON probes do all of their I/O over a network, there is no need for expensive display and disk systems on the probe, lowering the cost of an RMON probe to perhaps $1500–$3000. Only the management station needs to have good user interface capabilities, and this is where a windowing and multitasking UNIX or PC workstation can excel. Often the network manager can save money by loading RMON management station software onto an existing SNMP management station.

A protocol analyzer, on the other hand, performs both the data collection and the user interface tasks on the same system. Each of the several protocol analyzers in a similar environment must have a display, disk, and keyboard. This puts the protocol analyzer vendor between a rock and a hard place, choosing between a low-cost platform and a decent graphical user interface. As is typical in such a situation, neither function is served well, resulting in an expensive system with generation-old display capabilities. A typical system can cost $8,000–$20,000 for a single platform with a character-oriented display (proving, perhaps, that it is expensive to figure out how to shoehorn new features into such a limited platform). Most environments with a need to monitor more than a few nets will find an RMON solution increasingly attractive.

This cost savings allows the network administrator to make better decisions when designing a network management system. Rather than saving on scarce resources by sharing a protocol analyzer amongst several networks, it is possible to place a single RMON probe on each network for the same amount of money. Because each probe is dedicated to a network, they can pro-actively alert the network operations center when they detect an alarm. Trouble-shooting of a problem can begin instantly, without waiting to install a portable analyzer.

Better applications

Because there is little tendency to compromise on the capabilities of an RMON management station, the network manager can be more effective while using such a platform, and can typically see multiple networks in different windows at the same time. Likewise, because an RMON probe performs many tasks simultaneously, the network manager may perform different diagnostic functions simultaneously, without interrupting the gathering of long-term performance statistics. If there is more than one network manager, they may both access an RMON probe at the same time without affecting each other. In contrast, most protocol analyzers are based on single-tasking, single-user platforms.

One of the most significant advantages to RMON applications is that since RMON is an open standard, the network administrator may choose the best application products to fit a particular environment — independently of the choice of probe platform. This keeps the customer from being locked into a particular application, and allows vendors to create innovative software for different applications, from fault or performance monitoring to configuration management. A healthy choice of commercial and free software is forming.

Integration with other network management tools

Another advantage to the openness of RMON is that it can be easily integrated with other network management tools. It is trivial to access RMON data with currently available SNMP products, though much of the more complex RMON data requires more sophisticated interfaces to be truly useful. Most network configuration databases would benefit from a link to RMON for auto-discovery purposes. Performance analysis tools can gain a vantage point at the link layer in addition to monitoring routers and hosts. No longer is network data locked onto a single protocol analyzer platform.

The Future

RMON products will benefit greatly from the upcoming SNMP version 2 (SNMPv2). Because of its faster data downloading speed, high security and acknowledged alarms, SNMPv2 will enhance RMON greatly. A prototype has already been built and has demonstrated the improvements in RMON performance which come as a result of use with SNMPv2.

At present, the RMON MIB is standardized for Ethernets; but, the technical work for the Token Ring extensions for RMON has just been completed, so that

standard should be published soon. Shortly after the standard is published, Token Ring RMON products should be available.

Future extensions to RMON will include FDDI and the analysis of network layer protocols such as IP, IPX and AppleTalk. This analysis will enhance the configuration management capabilities of RMON and provide a source of information for performing fault management at these higher layers.

Protocol analyzers will continue to have a role as protocol decoders for network software developers. In addition, for environments with only a few nets, the simplicity of the self-contained protocol analyzer system may outweigh the cost savings of an RMON probe. This is especially true, given the growing number and sophistication of software-only protocol analyzers that transform a UNIX or PC workstation into a protocol analyzer — at great cost savings, and sometimes with more sophisticated graphical capabilities than their more expensive brothers. This solution will be most attractive when a dedicated system is not needed and a UNIX or PC platform is available.

There are many factors that will drive this trend toward remote monitoring with RMON and away from the stand-alone protocol analyzer model. The critical factors will be the cost savings available by the more effective use of resources RMON provides and the impact that the open RMON standard has upon the most important component of network management systems, the application.

Ask Dr. SNMP

Jeffrey D. Case

Dear *Dr. SNMP*,
I am confused by the plethora of terms related to SNMP. Would you explain the following: SNMP, SNMP Security, SMP, and SNMP version 2?
—*Too many Terms in Tokyo*

Dear *Too many Terms in Tokyo*,
I'm happy to provide the following brief taxonomy of SNMP terms for your use.

Back on the farm, we have a saying:

"Repetition is the key to learning."

Consequently, you may want to go over it several times. There will be an unannounced quiz next Monday.

SNMP version 1 was originally known as SNMP. The SNMP and the other components of the framework, including the Internet Standard SMI and related MIBs, were first developed in 1988. The SNMP framework continues to be the de facto and de jure standard network management framework of choice.

Enhancements to the SNMP version 1 framework to strengthen it in the areas of authentication, authorization, access control, privacy, and proxy relationships led to "SNMP Security". These proposed enhancements were developed over a period of approximately three years and published as Proposed Standards for the Internet community (RFCs 1351-1353) in early July, 1992. However, SNMP Security has been overtaken by events. These documents are being superseded by replacement documents as a part of the SNMP version 2 standardization effort described below. The consensus developed at the July 1992 IETF meeting was that SNMP Security would not be deployed as described in RFCs 1351-1353, but would be deployed simultaneously with SMP and SNMP version 2 in order to provide a single transition in the network management community for the benefit of both vendors and customers. Consequently, customers should not expect vendors to produce products based upon these documents which will be relegated to historical status in the near future. The few products based upon them are already obsolete because they cannot interoperate with SMP or SNMP version 2.

SMP is a comprehensive and detailed (approximately 200 pages) proposal for the evolution of the SNMP management framework. Among many other things, this proposal includes the enhancements described in the SNMP Security documents, with minor enhancements, corrections, clarifications, and simplifications. The SMP documents were the input to the SNMP version 2 standardization efforts. Several vendors demonstrated implementations of SMP on the show floor at INTEROP.

Finally, SNMP Version 2 is the output of an ongoing effort of two IETF working groups. The SNMPv2 Working Group was organized in September, met in October and again in November, and has completed most of its work. A parallel committee, which is focused on the necessary changes to the administrative framework, was organized in October, met in November, and is scheduled to meet again in December. The two committees share the goal of completing their work in late 1992 but there should be little surprise if it is not finished until early 1993.

Dear *Dr. SNMP*,
You have been openly critical of vendors who have chosen not to implement network control functions via SNMP sets. I think that is unfair. Our company chose to allow customers to monitor our products via SNMP but we believe the trivial authentication mechanisms found in SNMP are too weak to support control functions. We are afraid that an eavesdropper might learn the community

string and take control of the network. Our customers are able to telnet to the console to perform control functions. These operations are password protected.

— *Spreader from Sales*

Dear *Spreader from Sales*,
Down on the farm, we have a saying:

> "Spread the manure where you are going to grow the vegetables."

It seems to me that you are doing exactly that. It is understandable that you wish to minimize the loss of sales due to your company's failure to fully implement the protocol.

Having said that, it has never been clear to me why there are folks who think that using telnet with plain-text reusable passwords is somehow more than the security mechanisms in SNMP version 1. It reminds me of a story.

Two parents were traveling in the car and wanted to communicate with one another without the children in the back seat being able to eavesdrop. Whenever they came to a sensitive word, they spelled it out rather than saying it. This "security mechanism" is effective as long as the children are less than about four years old. Telnet is quite similar. The password is spread across multiple packets instead of a single packet. Similarly, this strategy is more secure than SNMP sets so long as the eavesdropper is less than about four years old.

Of course, the correct strategy is to fully implement the specification, including set commands, and let the customers decide if it is appropriate to deploy these functions in their networks, with the default factory settings disabling them. In any case, the strong security features of SNMP version 2 will soon be the norm and will eliminate any excuses your company has been spreading for its failure to be compliant.

Security and Protocols

Keith McCloghrie

In the previous issue of ***The Simple Times***, this column looked at the changes in the way management information is defined in the Simple Management Protocol (SMP) and Framework. That framework has since been accepted as the basis for SNMP version 2 (SNMPv2). In this issue, we'll look at: the changes in the protocol data units (PDUs), the new error codes and exceptions, and the new ways to specify compliance and conformance. In the next issue, we'll look at the changes to security procedures within the Administrative Framework.

Bulk Retrieval

One of the most exciting changes is the addition of the "awesome" `get-bulk` PDU, which requests the transfer of a potentially large amount of data. As well as being able to serve as a complete replacement for `get-next`, `get-bulk` also provides significant increases in the efficient and rapid retrieval of large tables such as a routing table with hundreds/thousands of entries or a bridge's forwarding database.

The basic concept of `get-bulk` is that, with one PDU, it requests multiple repeated `get-next` executions. Two parameters are included in the request: *non-repeaters* and *max-repetitions*. The max-repetitions parameter specifies the maximum number of repeated executions for (a subset of) the requested variables. In each repetition, the agent retrieves the variables and their values which lexicographically follow the variables retrieved by the previous repetition. The agent continues processing the repetitions until either the maximum number is reached, or else a maximum-sized `response` PDU is generated, whichever occurs first. The non-repeaters parameter specifies how many (if any) of the variables in the request are not subject to repeated executions. This is useful when the values of one or more scalars (e.g., `sysUpTime`) must be retrieved, along with variables from multiple rows of a table. The result is that only one copy of the `sysUpTime` is retrieved along with the multiple rows from the table.

Thus, through using `get-bulk`, a manager can retrieve in one request as many variables, e.g., from the rows of a large table, as will fit in a maximum-sized `response` without knowing the names, i.e., the instance identifiers, of the particular variables it wants to retrieve. When max-repetitions has a value of one, `get-bulk` operates identically to `get-next`, with the exception that `get-bulk` never returns a `tooBig` error; if a `get-next` would return `tooBig`, `get-bulk` will return less than a whole repetition.

Implementation experience with `get-bulk` has confirmed the very substantial reductions in both the number of requests and the elapsed time for large retrievals. The current "top-speed" record stands at retrieving over 9,300 tabular variables per second.

Other PDU Changes

SNMPv2 defines one other new PDU type: the `inform-request`. While some observers have described this as a "confirmed trap", it is important to note that the `inform-request` is not intended for use by agents. In fact, SNMPv2 retains the existing SNMP paradigm that transfers of management information should be initiated as close to the centers of management

control as possible. That is, transfers of management information should, in general, be initiated only when it is known that the receiver needs the information. In particular, traps which are asynchronously generated by an agent should not be sent unless it is known that the receiving management station wants them. (Consider, after a power outage in a building, most agents will generate `coldStart` traps even though the manager is completely aware that every system in the building has just rebooted.)

Rather, the `inform-request` is provided for use in manager-to-manager communication, where one manager has delegated a certain responsibility to another manager. Under these circumstances the manager-A requests manager-B to send InformRequests to notify it of specific conditions which may occur, and to retransmit such an `inform-request` for a defined number of times until the manager-A acknowledges it with a `response` PDU.

The other change to PDUs is a change in the format of the `trap` PDU, which has a header different from all PDUs. One of these differences is that it contains an address field in which only a TCP/IP address can be encoded. SNMPv2 fixes this by defining a new PDU, `snmpV2-trap`, which is identical in format to all other PDUs. Thus, addresses are now encoded in the `variable-bindings` field of the trap, where the address type can be properly identified. Another positive result of making all PDU formats identical is a reduction in the number of ASN.1 encoding/decoding routines needed by an implementation, thereby allowing a small reduction in code size.

Richer Error Codes and Exceptions

SNMPv2 defines twelve new error codes and introduces the notion of exceptions. The new error codes provide a much greater level of distinction between error conditions which occur on a `set-request`. For example, the new error codes, `inconsistentValue` and `resourceUnavailable`, are particular types of transient errors, whereas the new codes, `wrongValue` and `wrongLength`, indicate that the agent does not support either the value or the length of the specified value.

The three new exceptions are returned in response to retrieval requests on a per-variable basis (as opposed to error codes which are returned on a per-PDU basis). Thus, partial results of a retrieval can be returned with any exceptions flagged, rather than having an entire request rejected because of an error. Different exceptions allow a manager to distinguish between an object which is not implemented by an agent and an object for which no instances currently exist.

Improved Support for Sets

In addition to the additional error codes mentioned above, the operation of the `set-request` PDU is improved in several other ways. The authentication, authorization, and access control features of SNMP Security are incorporated into SNMPv2. The `TestAndIncr` textual convention can be used not only to ensure that multiple set operations are executed at most once and in the desired order, but also to provide advisory locking between multiple management applications. In addition, the `RowStatus` textual convention clarifies the procedures by which rows can be created and deleted from tables.

Compliance and Conformance

The SNMPv2 SMI specifies two macros for defining conformance and compliance: the module compliance macro and the agent capabilities macro.

The module compliance macro is used when describing requirements for agents with respect to managed objects. As mentioned in the last issue, object definitions in a MIB module define the maximal level of implementation which makes "protocol sense". In contrast, the module conformance macro defines the minimum requirements for conformance, specified in terms of objects within groups, where different groups may come from different MIB modules.

The agent capabilities macro is an evolution of a macro defined in RFC 1303. It may be used to describe, in a concise and machine parseable format, which MIB modules, objects, and values are actually implemented by a particular agent. When an agent capabilities macro is written, an OBJECT IDENTIFIER value is assigned to it. That assigned value is then used as the value of the `sysObjectID` object (defined in MIB-II) returned by that agent. Therefore, a management station which has preloaded a set of agent capabilities macros, can query an agent for its `sysObjectID` value, and can search its stored set of capabilities macros to find a match. If a match is found, the management station can then tailor its applications corresponding to the capabilities of that agent.

Coexistence and Transition

Much thought was given to an orderly evolution from SNMP to SNMPv2. It is expected that, despite the many benefits of SNMPv2, some time will elapse before the last SNMP system is upgraded or de-commissioned. During the period of co-existence, either or both of two approaches can be used: bilingual managers and the use of proxy.

Bilingual managers implement both versions of the SNMP. For example, a bilingual manager sends SNMP queries to SNMP agents and expects SNMP responses. Similarly, it sends SNMPv2 queries to SNMPv2 agents and expects SNMPv2 responses.

Alternatively, proxy may be used to convert from one message format to another. For example, a new management station might support only SNMPv2 message formats. These messages might be converted to SNMP messages via a proxy agent in order to support communications with older agents.

Both of these approaches were implemented early on so that feedback from the experience could be incorporated into the specification. Such feedback included the definition of the necessary MIB objects to identify the SNMP version 1 trap header information when carried in the variable-bindings of an SNMPv2 trap, and specifying that SNMPv2 managers are required to accept the old error codes even though there are no procedures in the SNMPv2 protocol specification which generate those old error codes.

There is a strong belief that the effort and emphasis placed on co-existence of and transition from SNMP to SNMPv2 will pay dividends as soon as the new systems implementing SNMPv2 start to become available.

Standards

David T. Perkins

In September and October, the Concise SMI (RFC 1212) was promoted to full standard status, and the 802.3 Repeater, X.25 LAPB, and X.25 PLP MIBs were published as proposed standards. The Ether-like MIB (RFC 1284) is in the process of being revised and promoted to draft standard status. The collection of documents defining SNMP over other transports (OSI, AppleTalk, and IPX), and the RIPv2 MIB are in the final stages of approval of being published as proposed standards.

At the November 1992 IETF meeting, several working groups (WGs) have put the final touches on MIBs and will be submitting them for consideration as standards. (See the *Working Group Synopses* column for the details.)

There are many MIBs at the proposed standard stage. They have been sitting there waiting for someone to champion the effort to move them to the next stage. This consists mainly of convincing developers to implement them, and then staging a testing of interoperability and completeness of several independently developed implementations. There is also a requirement that the implementations have seen adequate operational experience.

As a reminder, the standards process consists of three levels: proposed, draft, and full. The requirements for each stage were summarized earlier in ***The Simple Times*** (volume 1, number 2). RFC 1310 currently defines the standards process, which will likely be changing in 1993. Further, the standardization process may be radically changed due to changes in the interaction between the IAB, IESG, and IETF working groups as proposed by the POISED WG in the November 1992 IETF meeting. When a new process is established, this column will explain how it works.

Recently Published RFCs

RFC 1368 - IEEE 802.3 Repeater MIB (Proposed Standard)

This document defines the MIB objects needed for managing IEEE 802.3 repeaters, which are sometimes referred to as "hubs". The document is a straight-forward translation of the IEEE MIB. A long, but useful, introduction section explains the model and definition of terms. The MIB consists of a *basic* group which contains objects applicable to all repeaters. The optional *monitor* group contains objects for monitoring the repeater as a whole and for individual ports. The optional *address tracking* group contains objects that track the source MAC addresses seen on each port of the repeater. The MIB also defines three traps.

RFC 1381 - X.25 LAPB MIB (Proposed Standard)

The MIB objects for managing the X.25 link layer are defined in this document. Management of an X.25 protocol stack requires not only the objects from this MIB, but also the objects from the X.25 PLP (RFC 1382) and RS-232 Interface Type (RFC 1317) MIBs. The MIB is organized as four tables. The *admin* table contains common parameters used by LAPB to initialize interfaces. The *oper* table contains objects which reflect the monitored operational values of parameters of interfaces. The *Flow* table contains LAPB flow control statistics for each interface. Finally, the *XID* table, required only in those systems implementing XID negotiation, contains those needed parameters not found in the *admin* table.

RFC 1382 - X.25 PLP MIB (Proposed Standard)

This document defines the objects needed for managing the X.25 packet layer. This MIB contains many objects which are divided into seven tables. The *admin*, *oper*, and *stat* tables define the administrative objects, the operational objects, and statistics objects for each X.25 Packet Level Entity (PLE), respectively. The *channel*, *circuit*, *clearedCircuit*, and *callParm* tables contain the objects to configure channels, monitor current and abnormally terminated circuits, and specify or reflect the parameters for X.25 calls, respectively.

The Process used to define SNMP version 2

The Internet-standard Network Management Framework is defined by:

- 1155 - Structure of Management Information (SMI);
- 1157 - Simple Network Management Protocol (SNMP);
- 1212 - Concise MIB definitions; and,
- 1213 - Management Information Base (MIB-II).

which are all full Internet-standards. In addition, many MIBs make use of the notation defined in the informational RFC:

- 1215 - A convention for defining traps for use with the SNMP.

This framework, originally defined in 1987–8, and updated with the concise format and traps in 1990, has basically remained unmodified. With the exploding number of MIBs and operational experience, some rough edges were encountered.

At the August 1991 IETF meeting, a special session was held to gather a list of concerns and perceived deficiencies in the framework. Later that year at the December 1991 IETF meeting, the addition of security to the framework was proposed. In early 1992, preceding the March 1992 IETF meeting, a mail message was sent out to the IETF mailing list by the IETF's Area Director for Network Management, James R. (Chuck) Davin, which invited the submission of contributions that addressed the deficiencies in the SNMP framework. The call for proposals specified no timetable when the submission window would be closed — it left the closing to the time when sufficient number and quality had been submitted.

Two weeks prior to the July 1992 IETF meeting, the SNMP security documents were published as RFCs, and independently, a proposal for the new framework was submitted. This proposal, the *Simple Management Protocol (SMP) and Framework*, was made by Jeffrey D. Case, Keith McCloghrie, Marshall T. Rose, and Steven L. Waldbusser. The SMP proposal consisted of eight documents that ran to over 200 pages. The authors had also developed four independent, interoperable implementations. Very impressive!

A special session was held at the July 1992 IETF meeting to present the submission. The meeting was well-attended. After the presentation and discussion, a motion was made from the floor to move as quickly as possible to form a working group, and evaluate the documents and any other submissions. There was overwhelming consensus to this suggestion. Since the SMP proposal specified major changes to the just-published SNMP security documents, it was decided to table the existing SNMP security documents, evaluate the changes from the SMP proposal, but to continue to keep the security aspects in a separate working group. (The SNMP Security WG was given an extension — so no new group was formed.) The reason for this was simple: the new security proposals required an incompatible change in the SNMP message format, as did the SMP proposal — and the attendees overwhelmingly wanted a single transition.

A charter was put together for a new working group, the SNMPv2 WG, with Robert L. Stewart selected as chair and Marshall T. Rose selected as editor. A closing date of September 10, 1992 was specified for new proposals. The first working group meeting was scheduled for October with additional meetings, if needed, scheduled for the November 1992 IETF meeting, and the April 1993 IETF meeting, at which time the WG was challenged to be completed or face disbandment.

The first meeting was attended by approximately 30 SNMP diehards. The working group decided to accelerate the schedule and target a completion date of December 1992. A mid-December meeting was scheduled in case resolution could not be reached at the November IETF meeting. The members of this initial meeting thrashed around for the first day trying to resolve the scope of the framework. The fundamental question was "should information that would be useful only to management applications, and/or information useful only to agents be included in MIBs?" This question was not completely resolved at the meeting, but on the mailing list a consensus was building. Other technical issues were also appearing, such as adding creation and deletion operators. Worry was also expressed about getting the SNMP Security WG moving at the same pace so that the documents could be completed in parallel.

Five sessions (a record!) were allocated for the SNMPv2 WG at the November IETF meeting. This time there were over 200 attendees. Could progress be made here? After the first session, it looked like the set of topics would be exhausted since two large lists of issues had not been submitted yet. One list was by David T. Perkins, who had planned to complete it before the scheduled December meeting. After seeing that completion could be attained, and at the prompting of several key working group members, a list of 41 issues was generated overnight and thrashed out over three sessions the next day. The result was several changes, along with an ad hoc editing session the next morning to address some minor issues. (The results of this extra session were reported back to the working group later in the week.)

During the week, another large set of issues was sent in via electronic-mail. These issues were discussed and largely resolved on Friday morning. The working group set a deadline for dealing with the final unresolved issues via electronic-mail, and overwhelmingly decided that no meeting would be needed for December. The major issues now resolved, the mailing list could be used to tie up loose-ends. The updated documents were posted for review that evening with a four-week window available for discussion.

The SNMP Security WG proceeded quickly to adopt the changes from the SMP proposal. However, operational issues, due to proxies and timing aspects of manipulating MIB objects, appeared not to be adequately resolved by the SMP changes. The second session of the working group tackled these issues. A new proposal was presented by the SMP authors, which addressed these concerns, addressed the "party proliferation" problem, and simplified the Party MIB. However, an alternative approach was also presented which addressed the first set of issues. Not enough time was available to evaluate the proposals, so the working group decided to try to resolve the situation via electronic-mail; and, if unable to do so, then to use the meeting time slot reserved in December by the SNMPv2 WG. Finally, the SNMP Security WG was charged up to catch up with the SNMPv2 WG and finish by the end of the year.

In summary, this process to produce the next generation of the SNMP framework shows how dynamic and productive the IETF can be when challenged with a great problem. A self-selecting group of contributors puts together a complete solution, which is then submitted as a base solution which is reviewed and modified, where needed, through an open process. This is the IETF way for rapid development of technology. From the initial call for proposals to final Internet Drafts only nine months will have elapsed.

This issue's column has given a rambling description of the process that was used to evolve the Internet-standard Network Management Framework. The next issue will focus on techniques used by other WGs to accelerate the advancement of standards from Proposed to Draft status and determine if these techniques can be used for the SNMP MIBs.

Summary of Standards

Full Standards:

- 1155 - Structure of Management Information (SMI);
- 1157 - Simple Network Management Protocol (SNMP);
- 1212 - Concise MIB definitions; and,
- 1213 - Management Information Base (MIB-II).

Proposed Standards:

- 1229 - Extensions to the generic-interface MIB;
- 1230 - IEEE 802.4 Token Bus Interface Type MIB;
- 1231 - IEEE 802.5 Token Ring Interface Type MIB;
- 1232 - DS1 Interface Type MIB;
- 1233 - DS3 Interface Type MIB;
- 1239 - Reassignment of experimental MIBs to standard MIBs;
- 1243 - AppleTalk MIB;
- 1253 - OSPF version 2 MIB;
- 1269 - BGP version 3 MIB;
- 1271 - Remote LAN Monitoring MIB;
- 1284 - Ether-Like Interface Type MIB;
- 1285 - FDDI Interface Type MIB;
- 1286 - Bridge MIB;
- 1289 - DECnet phase IV MIB;
- 1304 - SMDS Interface Protocol (SIP) Interface Type MIB;
- 1315 - Frame Relay DTE Interface Type MIB;
- 1316 - Character Device MIB;
- 1317 - RS-232 Interface Type MIB;
- 1318 - Parallel Printer Interface Type MIB;
- 1351 - SNMP Administrative Model;
- 1352 - SNMP Security Protocols;
- 1353 - SNMP Party MIB;
- 1354 - SNMP IP Forwarding Table MIB;
- 1368 - IEEE 802.3 Repeater MIB;
- 1381 - X.25 LAPB MIB; and,
- 1382 - X.25 PLP MIB.

Experimental:

- 1187 - Bulk table retrieval with the SNMP;
- 1224 - Techniques for managing asynchronously generated alerts;

- 1227 - SNMP MUX protocol and MIB;
- 1228 - SNMP Distributed Program Interface (SNMP-DPI);
- 1238 - CLNS MIB;
- 1283 - SNMP over OSI; and,
- 1298 - SNMP over IPX.

Informational:

- 1147 - A network management tool catalog;
- 1215 - A convention for defining traps for use with the SNMP;
- 1303 - A convention for describing SNMP-based agents; and,
- 1321 - MD5 message-digest algorithm.

Historical:

- 1156 - Management Information Base (MIB-I).

Working Group Synopses

Robert L. Stewart

Although this issue's column doesn't cover the time span of the previous one, it includes considerable work on SNMP. Not counting progress in other working groups, the SNMPv2 working group, announced in the previous issue of ***The Simple Times***, substantially completed its work at the November 1992 IETF meeting, which occurred during the time this issue covers. (See the *Standards* column in this issue for a report on the overall operation of the SNMPv2 working group.) Here we'll cover some details of the discussions on the SNMPv2 mailing list, which played an important part in the quick progress and will be used to finish the work. Those discussions account for over half my collection of notes since my last column, so this issue will clearly hit only what I judge as the high points.

As usual, we'll start with the general-purpose SNMP mailing list, then go to the individual working group synopses, with SNMPv2 in its proper, alphabetical order.

SNMP General Discussion

The mailing list suffered a plague of messages with the subject "help". A well-meaning person on USENET had told their SNMP news group we were a *listserv* mailing list. A LOT of people were interested. The unwitting perpetrator apologized and published a correction, but we still have an occasional relapse.

An inquiry on managing modems with SNMP or CMIP found a company working on an SNMP MIB to fit with the Proposed Standard Character and RS-232 MIBs. Another respondent referred also to the DS1 and DS3 MIBs and suggested that the modem folks get together and work on it.

A question asked how to represent multiple logical interfaces in the `ifTable` received the suggestion of SMDS over DS1 and DS3 as separate `ifEntrys`, and the example of an implementation where Ethernet and 802.2 have two entries, identical except for type, along with the lament that this problem is big and growing.

A query whether a router should broadcast its `coldStart` trap to avoid loss due to lack of routes received the reply that broadcast is not allowed, but the router may choose to delay initialization until its view of the network stabilizes.

Unlike past outbreaks, the question of what to return if an agent doesn't support an entire MIB group received the simple reply that is non-conforming; RFC 1303 is how it is documented, and the agent should tell the truth and return `noSuchName`.

A self-admitted non-lawyer submitted a suggested copyright notice to consider for proprietary MIBs, clarifying ownership and granting a right to use.

An inquiry for standard or proprietary work on a data base MIB received no responses.

A person complained of having an implementation of the TCP connection table that has multiple entries with all zero indexes and state "closed", to which ISODE `snmpi` responds with an infinite loop, while another manager stops. The questioner asked if the answer is not to show entries or to append an extra index integer as suggested in *The Simple Book*. One reply stated that *The Simple Book* is out of date; the extra integer is a bad solution and the agent needs a more complete table in an enterprise MIB and then asked if we should fix this with a new table as exemplified by the new Forwarding Table MIB. Another reply pointed out that all zeroes for two connections makes no sense in TCP and should not appear in the MIB. Yet another reply stated that CIPSO security adds a security compartment, making five index values necessary, so that an agent could be confined to a security compartment or provide an additional index. The final word was that that sort of change will break all TCP code, so an SNMP MIB revision is trivial by comparison.

An inquiry how to formalize the definition of a trap that sends different lists of objects, depending on circumstances, received the reply that RFC 1215 defines the list of objects included in every instance of the trap, then says the agent can add others, but gives no way to list them.

An inquiry for implementations of the IP Forwarding Table MIB received no replies.

An inquiry for company-specific documentation on an asynchronous protocol for out-of-band, serial line access to SNMP drew considerable discussion and the documentation of one such protocol. Suggestions to form a working group, resulted in the suggestion to call a Birds of a Feather session (BOF), but that didn't happen.

A question whether a bridge should send `linkUp` and `linkDown` traps on spanning tree state changes became a debate on internal mechanisms, which evolved into a discussion of reliable traps, and ended in a shouting match, complete with circular arguments and personal attacks. The issue was not resolved.

Imbedded in the previously mentioned discussion was the suggestion to send an ICMP echo request before trying SNMP. Relating this to similar behavior on the INTEROP show network became a long discussion on management of that network and how it compares to managing enterprise networks. The consensus was that it is different but can offer useful insight.

The announcement of a Beholder BOF at INTEROP was followed by the immediate clarification that Beholder is a freely available network monitor.

The question of whether "public" is a well-known community with read-only access, and how to change it, brought the reply that it is a "should" in Router Requirements. Another response said it is simply a convention, considered good by some and bad by others, and the means of changing it is implementation-specific. Yet another response said there is no such convention, that Router Requirements speaks only for routers, and says nothing about how to change it.

An inquiry for the current list of enterprise numbers was referred to the file `mib/snmp-vendors-contacts` in the mib directory at `venera.isi.edu`.

A question about proper Network Management System (NMS) behavior in the face of different table columns missing depending on row type brought the response that NMSs that trim missing columns from subsequent requests would misbehave. This led to a discussion of fielded agents with bugs in `get-next` continuity and the suggestion that labs and press should report them. A statement that labs and press had blessed the implementations brought a reference to the blind leading the blind.

An inquiry for host or workstation MIB definitions was referred to the Host MIB Working Group.

Many inquiries for specific implementations or standards work received no reply.

A request for guidance in deciding whether to allow read-write access via SNMP received the response that it depends on the necessity to change values remotely, with considerations for security. The original requester clarified that the issue is not knowing what can be changed, but concern over loss of monitoring ability without write access.

An inquiry regarding progress on an ISDN MIB received the response there is some but it's very slow and a separate suggestion to restart a group that had been working on it.

BGP Working Group

Consensus on the BGP-4 MIB was assumed due to lack of discussion. Concerns over coexistence of BGP-4 and previous versions led to splitting some tables, following the November 1992 IETF meeting.

Bridge MIB Working Group

The MIB editor posted an announcement of a new draft, ready for Draft Standard, and included a list of changes. This was followed with various other minor changes and no further discussion, resulting in a new Internet Draft.

An inquiry for spanning tree test suites received the response that no official suite is known and the responder used a private set of configurations and tests.

A call for final review of the Internet Draft before recommendation to the IESG as a Draft Standard drew an objection to removal of the source routing group. The response explained that the Cambridge meeting separated source routing due to the number of changes continuing in IEEE 802.5, so the remainder of the MIB could advance. This drew the question whether the eventual source routing group would be entirely separate or part of the Bridge MIB. The answer that it would be in the same branch but a separate document got the response that such a disposition is acceptable but not desirable. The final status statement was that the source routing group has not been well reviewed; has no implementations, and could be a topic for the April 1993 IETF meeting. This brought a suggestion to write up the agreement so far and a separate statement that this is acceptable to the one party who had implemented the group.

Character MIB Working Group

A proposal on the SNMP mailing list to add an RS-232 MIB object to define the function of a port got the response that such an object is a good idea but the RS-232 MIB is not central enough, and the object should be in a revamped Interface group, so we're stuck with enterprise space for now. The original requestor admitted hoping to influence a smaller group, suggested that the Interface group be fixed, and agreed to use enterprise space.

Chassis MIB Working Group

Someone asked how information on newly installed cards is distributed and whether there is one master agent for the Chassis MIB. One response said there is no single method, the responder prefers an out-of-band channel and a central agent, suggested using down-loaded data, and said there should be one master agent per logical chassis. Another responder said the mechanism is implementation-specific and that an out-of band bus is best, citing an implementation that collects information dynamically, and pointed out there is no master point of control, allowing one or many chassis agents, but that all should supply the same information.

ASN.1 for power supplies and environmental sensors was submitted. Comments that a table used the power supply index but didn't include it, traps should be added, and Gauges can't represent negative numbers drew the responses that including indexes in the table is conventional but not required, traps should be subject to group consensus to disregard discouragement in RFC 1215, and Gauge values should have been INTEGERs.

Considerable discussion over the chassis model did not reach stable consensus, as was clearly shown at the November 1992 IETF meeting when the group could not reach a common understanding of the model. Various suggestions for changing or clarifying it were received. A new draft was available for the November 1992 IETF meeting, along with the editor's list of open issues.

DECnet Phase IV MIB Working Group

A comment that adjacency indexing is insufficient brought a quick response that this has been resolved and new implementations should use the new proposal.

The editor later proposed starting a new document, encouraged implementation, and stated a need for interoperability testing. A vendor responded they are shipping an implementation now, are agreeable with compatible changes, and are willing to test any time.

Domain Name Service Working Group

A final Internet Draft completing all work was announced. An objection to the model of name server versus resolver got the response that the model is based on RFC 1034 rather than particular implementations. The editor offered to clarify the model if its authors would supply text. Since three weeks passed with no comments, the draft is to be recommended as a Proposed Standard.

A request for a source of DNS statistics was referred to the MIB draft, which contains objects for the requested information.

Ethernet MIB Working Group

RFC 1369 was published, documenting implementation experience, and the IAB accepted the IESG's recommendation that the current Internet Draft be accepted as a Draft Standard.

FDDI MIB Working Group

A discussion regarding traps reached no conclusion.

A new draft, aligning with ANSI SMT version 7.2, was published, and the group met at the November 1992 IETF meeting.

Host MIB Working Group

An announcement of an October meeting in Pittsburgh, new drafts, an agenda, and two sessions scheduled for the November 1992 IETF meeting drew the objection that notice was too short for non-US people. This was agreed upon, with a promise to consider it for future meetings. The suggestion of an audio or videocast, based on other existing experience, brought a pledge to look into it and a solicitation of interest to help plan resources. Travel suggestions, times, directions, and agenda for the October meeting were posted.

A new draft brought many detailed questions and comments, such as an objection to removable media and caches in `hrStorageTable`, concern for the lack of specific problem information from some printers, a request for definition of a "page", and concern for the lack of information on network-mounted file systems. The editor agreed on removable media, added status for printers, requested ideas on "page", and said it is inappropriate to manage remotely-mounted disk drives other than at their own server, so the read-only list and location are sufficient. Another person suggested the MIB needs clarification of intent for `hrStorageVirtualMemory`, that counting pages doesn't support monitoring utilization, maintenance and refill needs, and that pages can be simple or complex but should be tuned to refill needs.

Minutes of the October meeting and an agenda for the November 1992 IETF meeting were published.

A query about coordination with the Desktop Management Task Force (DMTF) received multiple responses indicating considerable interaction.

IDPR Working Group

The agenda for the November 1992 IETF meeting included a discussion of MIB implementation. At the meeting, a developer stated the intention to implement much of the MIB.

A person who had implemented the MIB with `gated` offered many detailed comments and a revised MIB.

IPLPDN Working Group

A query concerning suitability of the Frame Relay MIB for customer management of a commercial service included many detailed questions. The single responder said the MIB is a good, minimum start but would need additional objects for interface description and switching issues, which are not covered due to implementation differences.

IS-IS Working Group

A new Integrated IS-IS MIB was announced. Another message sought implementations.

Multiport Repeater MIB Working Group

A query about handling multiple repeaters in a single agent got the response that this is handled by communities, parties, views, and the Chassis MIB. The followup question asking how to find the Chassis MIB was referred to its new draft and the Chassis MIB mailing list.

The MIB was published as RFC 1368.

The statement that there is no definition for an invalid `rptrAddrTrackLastSourceAddress` when the repeater hasn't seen an address included a suggestion to use a zero change counter as an indication. A response pointed out you can't return a null value, and the counter can wrap, and thus suggested an all-zero MAC address, then asked the working group if this should be added at Draft Standard time. Another message suggested that a zero counter and a zero address are low probability and should be the indicator as that allows a zero MAC address to be valid. This brought the response that a zero length address is best but that it requires deprecating the object.

NOCtools Working Group

An inquiry about an update to RFC 1147 got a reference to contents in individual files, available via anonymous FTP from `doc/noctools` at `wuarchive.wustl.edu`. This drew the comment that it should be easy to convert to a WAIS server.

A new draft was published, with a request for help to verify it.

OSPF Working Group

The complaint that the IP interface mask is missing from the OSPF MIB interface table got the response that it is available as MIB-II `ipAdEntNetMask`.

A new Internet Draft was published, followed by a last call on the MIB and traps, to be discussed at the November 1992 IETF meeting. At that meeting there were minor changes to the MIB and none to the traps.

A question on the `ospfIfTable`, whether an unnumbered interface's `ospfIfIpAddress` should have the value 0.0.0.0, received a simple yes and a concurring message.

PPP Working Group

The complaint that MRU negotiation doesn't necessarily have a correct result with ignorant peers, and that the MIB lacks ways to set remote MRU drew no response.

RIP Working Group

The MIB was recommended to the IAB as a Proposed Standard.

Remote Monitoring (RMON) MIB Working Group

The group planned separate meetings at the November 1992 IETF meeting, one for token ring and one for Ethernet.

Three separate, relatively detailed questions received no public response.

One message asked how an agent reports failures such as alarm generation, whether monitoring entries in dynamic table such as `tcpConnEntry` is allowed, whether `EntryStatus` is the only object that can be changed once an entry is validated, whether an entry becomes invalid if a bad parameter is set, what action to take if an entry is not acceptable but the NMS tries to set "valid", and how do you determine required parameters? One response answered that one implementation checks and invalidates alarms on failure, it invalidates the alarm if the dynamic entry goes away, some parameters can be changed on the fly, an invalid entry should remain to allow correction, and the implementation refuses to accept validation of an incomplete entry, insisting on definition of required parameters. This led to a long discussion of dependencies, responsibilities of NMS versus agent, and lack of guidance from RFC 1271, with no clear conclusion, other than the market will sort it out.

Various other individual questions received no public answer. Others received benefit of implementation experience.

A question on why the collector and interface are coupled for token ring but not for Ethernet, got the response that uncoupled configurations are not possible.

The announced goal for the November 1992 IETF meeting was to complete work on token ring extensions

for recommendation as Proposed Standard, and to review minor changes to RFC 1271.

An inquiry about FDDI RMON extensions got the reply that none exists.

SNMP Security Working Group

A complaint that SNMP Security is much too complex and a statement that SNMPv2 suggested the maximum practical revisions, drew a request for specific proposals, as the documents have had extensive review. The complainer responded as willing to let the work stand rather than introduce more delay, based on implementation experience, but regretted the greatly increased cost of entry into SNMP support.

A notice was posted regarding official government review of DES, with reference to consideration for software implementations.

Various concerns were expressed regarding work needed for SNMPv2 progress, getting the response that the work was planned for the November 1992 IETF meeting. The agenda came with the announcement of two meeting slots, a required reading list, the goal to complete work without a December meeting, and a list of topics. Another message added a few more topics.

A new charter for the SNMP Security working group, updated for SNMPv2, was published.

Based on user and vendor concerns over implementation cost, a proposal was submitted for compliance levels of "unSecurable", providing the new packet format with the same security as community-based SNMP; "unConfigurable", providing MD5 for authentication but no DES and no party creation (to minimize requirements for non-volatile storage); and, "full", including party creation, but with DES optional. A comment that the proposal was excellent included the argument that full compliance should require DES for secure party creation and proposed an additional key to secure key exchanges. The proposer agreed that DES is required for party creation but an additional key adds complexity where more simplicity is needed. A long discussion of optional security and use of XOR to change secrets without DES ensued. The XOR technique was well-defended by pointing out that it is no more or less secure than DES itself, in the face of compromised keys. Optional security was defended on the basis that security should be a network policy decision. It was attacked on the basis that it will result in insecure implementations and no net improvement in SNMP.

Following the November 1992 IETF meeting, various proposals from that meeting were submitted.

A proposal for Party MIB simplification was submitted. It received one message of immediate support, liking its three-party approach, simplified view table, and removal of complex indexing. Another message suggested that the single ACL table feature of limiting functions could be merged with the view table.

A proposal for eliminating the need for DES in party creation drew comments on the need for multiple secrets and particular secure implementation structures.

One co-chair stated that the group must resolve the issues of party proliferation and Party MIB compliance levels from the three proposals presented at the November 1992 IETF meeting, that the proposers will try to resolve their differences and present results to the list soon, and the group must try for consensus on the mailing list during that week, and, lacking consensus, the group will meet in Atlanta in December. One responder agreed. Another responder expressed concern over going too fast, and suggested that the group should just decide whether or not to meet, since cancelling does cost money. This drew support for holding the meeting in January to allow more time to consider. A co-chair reminded everyone that the charter calls for new documents in January and updates will take time after reaching consensus on the proposals; ignoring quality in favor of time would suggest throwing out the proposals, and that costs of cancelling are less than the cost of an unnecessary meeting. The host at Georgia Tech posted hotel suggestions.

SNMPv2 Working Group

The chair suggested that after some free discussion the group should accept a deadline for new issues and not allow others without overwhelming consensus.

A proposal to remove the one-hour 32-bit wrap restriction on the use of 64-bit counters received the response that some guidance is required to prevent overuse, and requested a better restriction. The suggestion not to use it unless the counter can get that big brought the response that any counter can get that big and that wrap time is the important question. Another message agreed that a restriction is necessary, as 64-bit operations in some implementations are very expensive.

A NAME clause for the OBJECT-TYPE macro was proposed to provide humans with better text than a textual descriptor. Over time, this proposal generated an incredible amount of discussion, with NMS writers favoring it and agent writers and MIB designers against it. Although consensus not to include it was apparently reached, the issue came back on the mailing list, resulting in confusion over consensus. Ultimately, it was discussed again at the November 1992 and rejected with the recommendation that it and related features be part of future work on information to improve the human

interface to NMSs.

A proposal for a way to compress object identifiers (OIDs) got much support and many objections, and also continued to come back in other forms, ultimately being put off for further study due to concern about cost versus value. That study did not occur, and the idea seemed to die, but came back twice more as proposals for optimizing the contents of `get-bulk` responses. It was rejected by the working group in that form as well, as an insufficient or too-complex mechanism to solve the level of problems being presented.

A proposal to liberalize the definition of a "manager", allowing subsets of manager functions, met concern over causing confusion with too many options. The proposer agreed to try a better definition but did not do so and the proposal died.

A proposal for a date and time textual convention drew considerable support and improvement on the mailing list and was accepted.

Proposals for a 64-bit integer and an unsigned integer met different fates.

The former was rejected at the November 1992 IETF meeting due to lack of a limiting function, and the latter was added.

The chair polled for problems with limiting textual descriptors to 64 characters, and none was voiced, so the proposal was accepted.

The chair sought strong objections to allowing changes in textual descriptors and enumeration names, and adding to enumerations. This topic generated considerable discussion and was deadlocked until resolved to not to allow changes to descriptors and names and to allow adding to enumerations.

Responding to a question, the chair indicated that people who miss meetings will have to present strong reasons for reconsidering a decision. Discussion ensued about concern for rushing versus quality work.

A discussion on strict requirements for implementation before recommendation wasn't settled until the November 1992 IETF meeting, where it was left at current implementations, which are more than required by normal IETF conventions.

A proposal for partial success status when a value could be set in volatile memory but not in non-volatile storage resulted in a long discussion and eventual consensus that the restart domain could handle the problem.

A consensus check on not-accessible objects led to very long discussions on whether index objects should be not-accessible to simplify agent implementations and optimize `get-bulk` responses. This polarized into NMS implementors who wanted a simple way to obtain such index objects and agent implementors who insisted that read-only indexes were not a solution. This ended up decided by the chair, leaving index objects not-accessible as originally proposed.

A call for consensus on time pressure resulted in concerns over rushing, but kept a December finish requirement. This was upheld at the November 1992 IETF meeting.

A proposal to pre-schedule additional meetings in January, February, and March, was rejected by the chair on the basis that this would encourage a lack of progress. There was no objection.

A discussion of defaulting deadlocked issues to either SNMPv1 or SNMPv2 ended up with the hope that all issues would be resolved with a "deadlock shelf" to hold issues for eventual resolution or disposal.

The chair's suggestion to accept no new proposals after November 9 was accepted with little discussion.

A proposal for adding explicit creation and deletion operations generated considerable discussion, pro and con, extended to the November 1992 IETF meeting. That meeting's consensus not to include them unraveled on the mailing list due to the proposers' inability to attend and their answering to objections. As agreed at the meeting, the issue must be settled by December 4. Tempers flared on the mailing list at the apparent disregard of the meeting consensus, but the angry parties quickly came to terms and made up.

The minutes for the November 1992 IETF meeting, and updated drafts, were available before midnight of the last meeting day, both being provided by the working group's editor.

A discussion of adding temporal semantics, that is, dealing with delays in the effect of operations, generated considerable discussion and is pending the output of the SNMP Security working group.

UPS MIB Working Group

A lengthy discussion about having separate MIBs for basic and advanced UPSs received the advice that a single MIB with required groups based on capabilities of the UPS is the conventional way to organize.

A query about lack of archived messages after the beginning of October got the response that the archiver was broken and the plan to rebuild from a private archive is in progress.

X.25 MIB Working Group

The X.25 LAPB MIB and X.25 PLP MIB are now Proposed Standards, RFCs 1381 and 1382, respectively. As a result, a few messages with suggested changes or problem resolutions were tabled for discussion at Draft Standard time.

SNMP Usage Survey

Jon Saperia of Digital Equipment Corporation is conducting a survey on SNMP and its uses in network management. The purpose of the survey is to help identify how people are currently using SNMP to monitor and manage their networks. The vendor or product line of management software is not of direct interest. What is sought is information that can be used by people who are involved in day-to-day management and planning of networks. In a future issue of ***The Simple Times***, the results of the survey will be presented.

Readers with electronic mail are encouraged to complete this survey and return it to:

`saperia@tcpjon.ogo.dec.com`

(For additional, ASCII copies of the survey, send a message to

`archive-server@simple-times.org`

and put

`mimesend simple-times/survey.1.5`

in the body.) Readers without electronic mail access can fax the completed survey to `+1 508-496-9929`, addressed to Jon Saperia at mailstop `OGO1-1/G17`.

I. Describe Yourself

1. Name.
2. E-Mail address.
3. Tasks you perform (e.g., network operations, software development).

II. Describe Your Network Environment

4. How many hosts are in your network environment?
5. What is the physical topology of your network environment (e.g., bridged, routed, WAN)?

> Give as much information as you can since there is probably a relationship between the environment and the type of tools used as well as frequently used MIB objects. If you are describing a regional, campus, corporate or other large net, include the name of the network.

6. What are the top 3 protocol suites used in your network environment (e.g., TCP/IP, Appletalk, DECNet, XNS)?

7. What are the top 4 problems that come up in keeping the network running properly that are related to the technology?

> Do not list non-technical issues, such as funding — at the end of this survey you may comment accordingly. Be as specific as you can, for example: "new router configurations always take time to get right".

8. What kinds of management tools do you use to fix your day-to-day problems?

> Be as specific as possible with the tool name and the type of problem, for example: "I use `traceroute` to identify unwanted routing changes." Please include non-SNMP based tools, as this will help determine to what degree SNMP does not yet meet certain needs.

III. Use of SNMP-based tools

9. List any SNMP-based software that you use.
10. Describe the 3 most common ways that you use the tools.

> This is similar to the question above, but it is restricted to SNMP-based tools. The purpose of this section is to learn how SNMP tools add uniquely to our ability to do fault, configuration, performance planning, and so on.

11. Which MIB objects are the most useful and why?

> How are these objects related, and what type of analysis, if any, is used? For example, "I compare `ifLastChange`, `ifOperStatus`, and `sysUpTime` on the interfaces of a router to determine if an intermittent problem is interface-specific or more general." Comment on whether you find enterprise-specific MIB objects helpful.

IV. The Future

12. What SNMP-based objects have not yet been defined and what would you like to use them for?
13. What SNMP applications are not available that you would like to help do fault, configuration, performance or other network management tasks?
14. Additional comments.

Recent Publications

Network Management: A Practical Perspective
Allan Leinwand and Karen Fang, Addison-Wesley, 1992.
ISBN 0–201–52771–5

Publication Information

The Simple Times is published with a lot of help from the SNMP community.

Publication Staff

Coordinating Editor:
Dr. Marshall T. Rose — Dover Beach Consulting, Inc.

Featured Columnists:
Dr. Jeffrey D. Case — SNMP Research, Inc.; University of Tennessee
Keith McCloghrie — Hughes LAN Systems, Inc.
David T. Perkins — SynOptics Communications, Inc.
Robert L. Stewart — Xyplex, Inc.
Steven L. Waldbusser — Carnegie Mellon University

Contact Information

Postal: ***The Simple Times***
c/o Dover Beach Consulting, Inc.
420 Whisman Court
Mountain View, CA 94043–2186

Tel: +1 415–968–1052
Fax: +1 415–968–2510
E-mail: st-editorial@simple-times.org
ISSN: 1060–6068

Submissions

The Simple Times solicits high-quality articles of technology and comment. Technical articles are refereed to ensure that the content is marketing-free. By definition, commentaries reflect opinion and, as such, are reviewed only to the extent required to ensure commonly-accepted publication norms.

The Simple Times also solicits terse announcements of products and services, publications, and events. These contributions are reviewed only to the extent required to ensure commonly-accepted publication norms.

Submissions are accepted only in electronic form. A submission consists of ASCII text. (Technical articles are also allowed to reference encapsulated PostScript figures.) Submissions may be sent to the contact address above, either via electronic-mail or via magnetic media (using either 8mm `tar` tape, 1/4in `tar` cartridge-tape, or 3-1/2in MS-DOS floppy-diskette).

Each submission must include the author's full name, title, affiliation, postal and electronic mail addresses, telephone, and fax numbers. Note that by initiating this process, the submitting party agrees to place the contribution into the public domain.

Subscriptions

The Simple Times is available via electronic-mail in three editions: *PostScript*, *MIME* (the multi-media 822 mail format), and *richtext* (a simple page description language). For more information, send a message to

```
st-subscriptions@simple-times.org
```

with a `Subject` line of

```
help
```

In addition, ***The Simple Times*** has numerous hard-copy distribution outlets. Contact your favorite SNMP vendor and see if they carry it. If not, contact the publisher and ask for a list. (Communications via e-mail or fax are preferred).