IETF SNMP Working Group
Internet Draft
SNMP Communications Services

April 1991

Frank J. Kastenholz

1. Status of This Memo

This Internet Draft document will be submitted to the RFC
editor for publication as an Informational RFC. The following
is proposed as the status paragraph of the published RFC:

> This RFC is being distributed to members of the Inter-
> net community as an Informational RFC. The intent is
> to present a discussion on the issues relating to the
> communications services for SNMP. While the issues
> discussed may not be directly relevant to the research
> problems of the Internet, they may be interesting to a
> number of researchers and implementors.

Distribution of this memo is unlimited.

2. Introduction

This document discusses various issues to be considered when
determining the underlying communications services to be used
by an SNMP implementation.

As reported in RFC 1052, IAB Recommendations for the
Development of Internet Network Management Standards [8], a
two-prong strategy for network management of TCP/IP-based
internets was undertaken. In the short-term, the Simple
Network Management Protocol (SNMP), defined in RFC 1067, was
to be used to manage nodes in the Internet community. In the
long-term, the use of the OSI network management framework was
to be examined. Two documents were produced to define the
management information: RFC 1065, which defined the Structure
of Management Information (SMI), and RFC 1066, which defined
the Management Information Base (MIB). Both of these
documents were designed so as to be compatible with both the
SNMP and the OSI network management framework.

This strategy was quite successful in the short-term:
Internet-based network management technology was fielded, by
both the research and commercial communities, within a few
months. As a result of this, portions of the Internet
community became network manageable in a timely fashion.

In May of 1990, the core documents were elevated to "Standard
Protocols" with "Recommended" status. As such, the Internet-
standard network management framework consists of: Structure
and Identification of Management Information for TCP/IP-based
internets, RFC 1155 [9], which describes how managed objects
contained in the MIB are defined; Management Information Base
for Network Management of TCP/IP-based internets, which
describes the managed objects contained in the MIB, RFC 1156
[10]; and, the Simple Network Management Protocol, RFC 1157
[1], which defines the protocol used to manage these objects.

In parallel with this activity, documents specifying how to
transport SNMP messages over protocols other than UDP/IP have
been developed and published: SNMP Over Ethernet [3], SNMP
Over OSI [2], and SNMP Over IPX [6] and it would be suprising
if more were not developed. These memos have caused a degree
of confusion in the community. This document is intended to

disperse that confusion by discussing the issues of relevance
to an implementor when choosing how to encapsulate SNMP

packets.

None of these documents have been made full Internet Standards. SNMP Over ISO and SNMP Over Ethernet are both Experimental protocols. SNMP Over IPX [6] is an Internet Draft. Only the SNMP Specification [1] is an Internet Standard.

No single transport scheme can be considered the absolute best solution for all circumstances. This note will argue that, except for a very small set of special circumstances, operating SNMP over UDP/IP is the optimal scheme.

This document does not present a standard or a protocol for the Internet Community. For production use in the Internet the SNMP and its required communication services are specified in [1].

3. Standardization

Currently, the SNMP Specification [1] only specifies that the UDP protocol be used to exchange SNMP messages. While the IAB may standardize other protocols for use in exchanging SNMP messages in the future, only UDP is currently standardized for this purpose.

In order to claim full compliance with the SNMP Specification, an implementation would have to use UDP for SNMP message exchange.

4. Interoperability

Interoperability is the degree to which the equipment produced by one vendor can can operate with equipment produced by another vendor.

Related to Interoperability is compliance with a standard. Everything else being equal, a device that complies with some standard is more likely to be interoperable than a device that does not.

For commercial product development, the pros and cons of
developing an interoperable product must be weighed and a
choice made. Both engineering and marketing organizations
participate in this process.

The Internet is the single largest market for SNMP systems. A
large portion of SNMP systems will be developed with the
Internet as a target environment. Therefore, it may be
expected that the Internet's needs and requirements will be
the driving force for SNMP. SNMP over UDP/IP is specified as
the "Internet Standard" protocol. Therefore, in order to
operate in the Internet and be managed in that environment on
a production basis, a device must support SNMP over UDP/IP.
This situation will lead to SNMP over UDP/IP being the most
common method of operating SNMP. Therefore, the widest degree
of interoperability and widest acceptance of a commercial
product will be attained by operating SNMP over UDP/IP.

The preponderance of UDP/IP based network management stations
also strongly suggests that an agent should operate SNMP over
UDP/IP.

The results of the interoperability decision drive a number of
technical decisions. If interoperability is desired, then SNMP
must be operated over UDP/IP.

5. To Transport or Not To Transport

A major issue is whether SNMP should run on top of a
transport-layer protocol (such as UDP) or not. Typically, the
choice is to run over a transport/network/data link protocol
or just run over the datalink. In fact, several protocols
have been published for operating SNMP over several different
datalink and transport protocols.

Operation of SNMP over a Transport and Network protocol stack
is preferred. These protocols provide at least five functions
that are of vital importance to the movement of SNMP packets
through a network:

o Routing
     The network layer provides routing functions, which
     improves the overall utility of network management. The

network has the ability to re-route packets around failed
areas. This allows network management to continue
operating during localized losses of service (It should
be noted that these losses of service occur not only
because of failures, but also for non-failure reasons
such as preventive maintenance).

o Media Independence
     The network layer provides a high degree of media
     independence. By using this capability, many different
     types of network elements may be managed. Tying SNMP to
     a particular data link protocol limits the management
     scope of those SNMP entities to just those devices that
     use that datalink protocol.

o End-to-End Checksum
     The end-to-end checksum provided by transport protocols
     improves the reliability of the data transfer.

o Multiplexing/Demultiplexing
     Transport protocols provide multiplexing and
     demultiplexing services. These services facilitate the
     many-to-many management relationships possible with SNMP.

o Fragmentation and Reassembly
     This is related to media independence. IP allows SNMP
     packets to transit media with differing MTU sizes. This
     capability is not available for datalink specific
     transmission schemes.

     Fragmentation and Reassembly does reduce the overall
     robustness of network management since, if any single
     fragment is lost along the way, the operation will fail.
     The worse the network operates, the higher the
     probability that a fragment will get lost or delayed.
     For monitoring and data gathering while the network is
     operating normally, Fragmentation and Reassembly is not a
     problem. When the network is operating poorly (and the
     network operators are typically trying to diagnose and
     repair a failure), small packets should to be used,
     preventing the packet from being fragmented.

There are other services and functions that are provided by a
connection oriented transport. These services and functions
are not desired for SNMP. A later section will explore this

issue in more detail.

The main drawbacks that are cited with respect to using
Transport and Network layers in the managed object are: a)
Increased development time and b) Increased resource
requirements. These arguments are less than compelling.

There are several excellent public domain or freely
redistributable UDP/IP stacks that provide enough support for
SNMP. The effort required to port the essential components of
one of these stacks is small compared to the overall effort of
installing the SNMP software.

The additional resources required in the managed object to
support UDP/IP are minimal. CPU resources are required only
when actually transmitting or receiving a packet. The largest
single resource requirement of a UDP/IP is calculating the UDP
checksum, which is very small compared to the cost of doing
the ASN.1 encoding/decoding, Object Identifier lookup, and so
on.

The author has personal knowledge of a UDP/IP stack that was
developed expressly for the purpose of supporting SNMP. This
stack requires less than 4Kb of code space. It is a
minimalist implementation of UDP/IP in that it is "just
enough" so support SNMP. This stack supports UDP, IP, ARP, and
handles ICMP redirect and echo request messages. Furthermore,
this stack was developed by a single person in approximately
two months. Obviously, neither the development effort nor the
memory requirements are large.

The network overhead of using UDP/IP is relatively small. A
UDP/IP header requires 28 octets (assuming no IP options).
Since the UDP is connectionless, it will generate no overhead
traffic of its own (such as TCP SYNs, FINs, and ACKs).

The growing popularity of internetworking outside of The
Internet mandates that SNMP operate over, at least, a network
layer protocol. These internetworks consist of a number of
networks all connected together with routers. In order to
traverse a router, a packet must be one of the network layer
protocols that the router understands. Therefore, for SNMP
management to be deployed in an internetwork, the SNMP
entities in that internetwork must use a network layer

protocol. SNMP over a datalink can not traverse a router.

There are some circumstances where running SNMP over some
datalink is appropriate.

There are schemes are under development to provide Out-Of-Band
(OOB) management access to network devices. This OOB access
is typically provided over point-to-point or dial-up
connections. Since these connections are dedicated to OOB
network management and go directly from the network management
station to the managed device, a Transport/Network protocol
may not be necessary.

Using a Transport/Network protocol on these links may be
easier from a development point of view though. It is
probably a simple configuration operation to have the
management station's IP use a serial port rather than the
"normal" (e.g., Ethernet) port for traffic destined for a
particular node.

If the Out-Of-Band link is also used as a "primary" route to
some nodes, then the functions of a network-layer are
required. These functions are readily supplied by using
UDP/IP.

For a datalink interface and driver (e.g., a PC Ethernet
interface card) that must be manageable independent of the
higher level protocol suite (which might NOT be manageable),
operating SNMP directly over the datalink is reasonable. It
is not known, a priori, what higher-level protocol services
may be available, so those services can not be used. If an
arbitrary choice is made for example, to put in an elementary
UDP/IP stack, then there may be two independent UDP/IPs in the
system (which is undesireable as this would require two IP
addresses per managed node), or a new protocol stack will be
introduced into the environment.

6. Connection Oriented vs. Connectionless

While this section primarily addresses itself to transport
layer issues, its basic discussion of connection oriented vs
connectionless applies to any layer which provides

communication services for SNMP.

For SNMP, connectionless transport service (UDP) is specified
in the Protocol Specification [1]. This choice was made after
careful study and consideration by the original SNMP
developers.

The prime motivation of this choice is that SNMP must continue
to operate (if at all possible) when the network is operating
at its worst. For other applications, such as Telnet or FTP,
the user can always "try again later" if the network is
operating poorly. On the other hand, the major purpose of a
network management protocol is to fix the network when it is
operating poorly so the "try again later" strategy is useless.

By using a connectionless transport protocol, SNMP takes on
the responsibility of reliable data transmission (A SNMP
application may time out outstanding requests and either
retransmit them or abort them as appropriate). However, the
SNMP requires these functions only of the sender of a Request
PDU (get, getNext, or Set), which typically is a network
management station. Since the Agent only generates responses,
it need not perform any of these functions. This vastly
reduces the resource and functional requirements on the Agent.

If a connection oriented transport is used, then a fundamental
design choice must be made with respect to connection
maintenance:.

(1) Keep a connection open to each managed object on the
     network,

(2) Establish and tear down connections on a per-operation
     basis, or

(3) Keep a fixed number of connections open and, when another
     connection is needed, use some algorithm (e.g., LRU) to
     select one for closing and opening to the new agent.

All of these alternatives pose severe problems, and because of
them, each is undesirable.

The first option reduces the amount of resources required to
perform a single operation in that the connection
establishment and termination cost is "amortized" over many

operations. On the other hand, keeping a connection open
implies that the management station needs to maintain a large
number of connection records (in the hundreds or even
thousands). Furthermore, if either side of the connection
engages in "keep-alives" (even though such behavior is frowned
upon), a large amount of traffic will be generated, consuming
a large amount of network resources, all for no gain.

The second option reduces the amount of idle resources such as
connection records, but vastly increases the amount of
resources required to perform an operation. A connection must
be established, the request Message sent and the response
returned, and then the connection closed for each operation.
For a TCP, this would typically require 10 separate packet
transfers plus the TCP Time-Wait (see the Appendix for
details).

In the face of pathological network problems, a connection
oriented transport protocol may simply cease to operate
because the probability of getting all of these packets
through reduces to a very small number.

The third option requires that the management station maintain
connection usage information in order to implement the LRU
algorithm. This excessively complicates the management
station. Furthermore, this option tends to reduce to the
second option when doing health check polling for a number of
agents that is large compared to the number of supported
connections.

A connection oriented transport protocol may provide services
that are undesirable or unneeded by SNMP.

For example, one application of network management is to poll
nodes to determine if they are up or not. When a node is up,
it makes little difference whether SNMP operates over TCP or
UDP. However, if the node goes down then TCP will eventually
close the connection. Every poll request must then be
translated into a TCP Open request while the managed node is
down. Once the node comes up, the send must then be done.

For connection oriented transports, the transport ACK does not
necessarily indicate delivery of data to the destination
application process (for TCP, see section 2.6 of [4]). The

SNMP would still need its own timeout/retry procedure to
ensure that the SNMP software actually got the packet.

A connection oriented transport such as TCP provides flow
control for the data stream. Because of the lock-step nature
of SNMP protocol exchanges, this is not a service that SNMP
requires.

Architectural purists may argue that an "Application" layer
entity (SNMP) must not perform operations that are properly
the realm of the Transport layer (timeouts, retransmissions,
and so on). While architecturally pure, this line of
reasoning is not relevant. The network management
applications and protocols are monitoring the "health" of the
network and, as a result, have the best information and are in
the best position to adapt their own behavior to the state of
the network, and thereby, continuing operations in the face of
network adversity.

7. Which Protocol

The final point of discussion is the actual choice of a
protocol to support SNMP.

If a device is destined for use in the Internet then it must
operate SNMP over UDP/IP.

If the device is operating in some other protocol environment,
then SNMP ought to use the transport services that are native
to that environment. It may make very little sense to
introduce a new protocol stack into a network in order to
provide just one service. For example, it could require that
the network operations staff understand and be able to
administer and operate two protocol stacks, that hosts and
routers understand both protocols, and so on. It may also be
bureaucratically impossible to introduce UDP/IP into the
environment (the "We are only a FOONET shop - if it doesn't
speak FOONET, we don't want it" argument).

References [2] and [6] are experimental standards for
operating SNMP over IPX and OSI respectively, In these
environments, those standards ought to be adhered to.

8. Security Considerations

Security issues are not discussed in this memo.

9. Appendix

This appendix details the TCP packet transfers required to
perform a single SNMP operation assuming that the connection
is established only for that operation and that a single SNMP
operation (e.g., get request) is performed. We also assume
that all operations are "normal" i.e., that there are no lost
packets, no simultaneous opens, no half opens, and no
simultaneous closes. We also ignore the possibility of TCP
segmentation and IP fragmentation.

The nomenclature used to illustrate the packet transactions is
the same as that used in the TCP Specification [4].


```
     Packet  Management                            Managed
     Number  Station                               Object
                     Connection Open...
       1          >--<CTL=SYN>----------------------->
       2          <--<CTL=SYN,ACK>-------------------<
       3          >--<CTL=ACK>----------------------->
                    Connection now open,
                    SNMP Request is sent.
       4          >--<DATA=SNMP Request>------------->
                    Response comes back
       5          <--<DATA=SNMP Response, CTL=ACK>---<
       6          >--<CTL=ACK>----------------------->
                    Operation is complete,
                    Management station initiates the
                    close.
       7          >--<CTL=FIN,ACK>------------------->
       8          <--<CTL=ACK>----------------------->
       9          <--<CTL=FIN,ACK>-------------------<
      10          >--<CTL=ACK>----------------------->
                    Wait 2 MSL
                    Connection now closed.
```

Some optimizations are possible IF the TCP has knowledge of
the type of operation. However, a general purpose TCP would
not be tuned to SNMP operations so those optimizations would
not be done.

10. References

[1] Case, J., M. Fedor, M. Schoffstall, and J. Davin, A
     Simple Network Management Protocol, Internet Working
     Group Request for Comments 1157, Network Information
     Center, SRI International, Menlo Park, California, May
     1990.

[2] Rose, M.T., SNMP over OSI, Internet Working Group Request
     for Comments 1161, Network Information Center, SRI
     International, Menlo Park, California, June 1990.

[3] Schoffstall, M.L., Davin, C., Fedor, M., Case, J.D., SNMP
     over Ethernet Internet Working Group Request for Comments
     1089, Network Information Center, SRI International,
     Menlo Park, California, February, 1989.

[4] Postel, J.B., Transmission Control Protocol Internet
     Working Group Request for Comments 793, Network
     Information Center, SRI International, Menlo Park,
     California, September 1981.

[5] Postel, J.B. User Datagram Protocol Internet Working
     Group Request for Comments 768, Network Information
     Center, SRI International, Menlo Park, California, August
     1980.

[6] Wormley, R., SNMP Over IPX, Internet Draft, August 1990.

[7] Defense Advanced Research Projects Agency, Internet
     Activities Board. IAB Official Protocol Standards
     Internet Working Group Request for Comments 1140, Network
     Information Center, SRI International, Menlo Park,
     California, May 1990.

[8] V. Cerf, IAB Recommendations for the Development of
     Internet Network Management Standards. Internet Working
     Group Request for Comments 1052. Network Information

        Center, SRI International, Menlo Park, California,
        (April, 1988).

[9] M.T. Rose and K. McCloghrie, Structure and Identification
     of Management Information for TCP/IP-based internets,
     Internet Working Group Request for Comments 1155.
     Network Information Center, SRI International, Menlo
     Park, California, (May, 1990).

[10] K. McCloghrie and M.T. Rose, Management Information Base
     for Network Management of TCP/IP-based internets,
     Internet Working Group Request for Comments 1156.
     Network Information Center, SRI International, Menlo
     Park, California, (May, 1990).

11. Acknowledgements

The author wishes to thank Jeff Case, Chuck Davin and Keith
McCloghrie for their technical and editorial contributions to
this document.

12. Author's Address

Frank Kastenholz
Clearpoint Research Corporation
35 Parkwood Drive
Hopkinton Mass 01748
Phone: (508)435-2000
Email: kasten@europa.clearpoint.com

Table of Contents