SNMP over OSI

Status of this Memo

This memo defines an Experimental Protocol for the Internet community. Discussion and suggestions for improvement are requested. Please refer to the current edition of the "IAB Official Protocol Standards" for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Table of Contents

1. Background ............................................ 1
   1.1 A Digression on User Interfaces .................... 2
   1.1.1 Addressing Conventions for UDP-based service .... 3
   1.2 A Digression of Layering ............................ 3
   2. Mapping onto CLTS .................................... 3
   2.1 Addressing Conventions ............................. 4
   2.1.1 Conventions for CLNP-based service ............... 4
   3. Mapping onto COTS .................................. 4
   3.1 Addressing Conventions ............................ 5
   3.1.1 Conventions for TP4/CLNP-based service .......... 5
   3.1.2 Conventions for TP0/X.25-based service .......... 6
   4. Trap PDU ............................................. 6
   5. Acknowledgements .................................... 7
   6. References .......................................... 7
   7. Security Considerations................................ 8
   8. Author’s Address..................................... 8

1. Background

The Simple Network Management Protocol (SNMP) as defined in [1] is now used as an integral part of the network management framework for TCP/IP-based internets. Together, with its companions standards, which define the Structure of Management Information (SMI) [2], and the Management Information Base (MIB) [3], the SNMP has received widespread deployment in many operational networks running the Internet suite of protocols.

It should not be surprising that many of these sites might acquire OSI capabilities and may wish to leverage their investment in SNMP technology towards managing those OSI components. This memo addresses these concerns by defining a framework for running the SNMP...
in an environment which supports the OSI transport services.

In OSI, there are two such services, a connection-oriented transport services (COTS) as defined in [4], and a connectionless-mode transport service (CLTS) as defined in [5]. Although the primary deployment of the SNMP is over the connectionless-mode transport service provided by the Internet suite of protocols (i.e., the User Datagram Protocol or UDP [6]), a design goal of the SNMP was to be able to use either a CO-mode or CL-mode transport service. As such, this memo describes mappings from the SNMP onto both the COTS and the CLTS.

1.1. A Digression on User Interfaces

It is likely that user-interfaces to the SNMP will be developed that support multiple transport backings. In an environment such as this, it is often important to maintain a consistent addressing scheme for users. Since the mappings described in this memo are onto the OSI transport services, use of the textual scheme described in [7], which describes a string encoding for OSI presentation addresses, is recommended. The syntax defined in [7] is equally applicable towards transport addresses.

In this context, a string encoding usually appears as:

```
[<t-selector>/]<n-provider><n-address>[+<n-info>]
```

where:

1. `<t-selector>` is usually either an ASCII string enclosed in double-quotes (e.g., "snmp"), or a hexadecimal number (e.g., '736e6d70'H);

2. `<n-provider>` is one of several well-known providers of a connectivity-service, one of: "Internet=" for a transport-service from the Internet suite of protocols, "Int-X25=" for the 1980 CCITT X.25 recommendation, or "NS+" for the OSI network service;

3. `<n-address>` is an address in a format specific to the `<n-provider>`; and,

4. `<n-info>` is any additional addressing information in a format specific to the `<n-provider>`.

It is not the purpose of this memo to provide an exhaustive description of string encodings such as these. Readers should consult [7] for detailed information on the syntax. However, this
memo recommends that, as an implementation option, user-interfaces to
the SNMP that support multiple transport backings SHOULD implement
this syntax.

1.1.1. Addressing Conventions for UDP-based service

In the context of a UDP-based transport backing, addresses would be
encoded as:

    Internet=<host>+161+2

which says that the transport service is from the Internet suite of
protocols, residing at <host>, on port 161, using the UDP (2). The
token <host> may be either a domain name or a dotted-quad, e.g., both

    Internet=cheetah.nyser.net+161+2

and

    Internet=192.52.180.1+161+2

are both valid. Note however that if domain name "cheetah.nyser.net"
maps to multiple IP addresses, then this implies multiple transport
addresses. The number of addresses examined by the application (and
the order of examination) are specific to each application.

Of course, this memo does not require that other interface schemes
not be used. Clearly, use of a simple hostname is preferable to the
string encoding above. However, for the sake of uniformity, for
those user-interfaces to the SNMP that support multiple transport
backings, it is strongly RECOMMENDED that the syntax in [7] be
adopted and even the mapping for UDP-based transport be valid.

1.2. A Digression of Layering

Although other frameworks view network management as an application,
extensive experience with the SNMP suggests otherwise. In essence,
network management is a function unlike any other user of a transport
service. The citation [8] develops this argument in full. As such,
it is inappropriate to map the SNMP onto the OSI application layer.
Rather, it is mapped to OSI transport services, in order to build on
the proven success of the Internet network management framework.

2. Mapping onto CLTS

Mapping the SNMP onto the CLTS is straight-forward. The elements of
procedure are identical to that of using the UDP, with one exception:
a slightly different Trap PDU is used. Further, note that the CLTS
and the service offered by the UDP both transmit packets of information which contain full addressing information. Thus, mapping the SNMP onto the CLTS, a "transport address" in the context of [1], is simply a transport-selector and network address.

2.1. Addressing Conventions

Unlike the Internet suite of protocols, OSI does not use well-known ports. Rather demultiplexing occurs on the basis of "selectors", which are opaque strings of octets, which have meaning only at the destination. In order to foster interoperable implementations of the SNMP over the CLTS, it is necessary define a selector for this purpose.

2.1.1. Conventions for CLNP-based service

When the CLTS is used to provide the transport backing for the SNMP, demultiplexing will occur on the basis of transport selector. The transport selector used shall be the four ASCII characters

\texttt{snmp}

Thus, using the string encoding of [7], such addresses may be textual, described as:

\texttt{"snmp"/NS+<nsap>}

where:

(1) \texttt{<nsap>} is a hex string defining the nsap, e.g.,

\texttt{"snmp"/NS+4900590800200038bafe00}

Similarly, SNMP traps are, by convention, sent to a manager listening on the transport selector

\texttt{snmp-trap}

which consists of nine ASCII characters.

3. Mapping onto COTS

Mapping the SNMP onto the COTS is more difficult as the SNMP does not specifically require an existing connection. Thus, the mapping consists of establishing a transport connection, sending one or more SNMP messages on that connection, and then releasing the transport connection. Further, a slightly different Trap PDU is used.
Consistent with the SNMP model, the initiator of a connection should not require that responses to a request be returned on that connection. However, if a responder to a connection sends SNMP messages on a connection, then these MUST be in response to requests received on that connection.

Ideally, the transport connection SHOULD be released by the initiator, however, note that the responder may release the connection due to resource limitations. Further note, that the amount of time a connection remains established is implementation-specific. Implementors should take care to choose an appropriate dynamic algorithm.

Also consistent with the SNMP model, the initiator should not associate any reliability characteristics with the use of a connection. Issues such as retransmission of SNMP messages, etc., always remain with the SNMP application, not with the transport service.

3.1. Addressing Conventions

Unlike the Internet suite of protocols, OSI does not use well-known ports. Rather demultiplexing occurs on the basis of "selectors", which are opaque strings of octets, which have meaning only at the destination. In order to foster interoperable implementations of the SNMP over the COTS, it is necessary define a selector for this purpose. However, to be consistent with the various connectivity-services, different conventions, based on the actual underlying service, will be used.

3.1.1. Conventions for TP4/CLNP-based service

When a COTS based on the TP4/CLNP is used to provide the transport backing for the SNMP, demultiplexing will occur on the basis of transport selector. The transport selector used shall be the four ASCII characters

\[ \text{snmp} \]

Thus, using the string encoding of [7], such addresses may be textual, described as:

"snmp"/NS+<nsap>

where:

(1)  <nsap> is a hex string defining the nsap, e.g.,

"snmp"/NS+4900590800200038bafe00
Similarly, SNMP traps are, by convention, sent to a manager listening on the transport selector

```
    snmp-trap
```

which consists of nine ASCII characters.

### 3.1.2. Conventions for TP0/X.25-based service

When a COTS based on the TP0/X.25 is used to provide the transport backing for the SNMP, demultiplexing will occur on the basis of X.25 protocol-ID. The protocol-ID used shall be the four octets

```
    03018200
```

This is the X.25 protocol-ID assigned for local management purposes. Thus, using the string encoding of [7], such addresses may be textual described as:

```
    Int-X25=<dte>+PID+03018200
```

where:

1. `<dte>` is the X.121 DTE, e.g.,

```
    Int-X25=23421920030013+PID+03018200
```

Similarly, SNMP traps are, by convention, sent to a manager listening on the protocol-ID

```
    03019000
```

This is an X.25 protocol-ID assigned for local purposes.

### 4. Trap PDU

The Trap-PDU defined in [1] is designed to represent traps generated on IP networks. As such, a slightly different PDU must be used when representing traps generated on OSI networks.

```text
RFC1283 DEFINITIONS ::= BEGIN
IMPORTS
    TimeTicks
    VarBindList
FROM RFC1157-SNMP -- [1] --
    ClnpAddress
```
FROM CLNS-MIB     -- [9] --;

Trap-PDU ::=  
[4]
IMPLICIT SEQUENCE {  
enterprise              -- type of object generating  
OBJECT IDENTIFIER,  -- trap, see sysObjectID
agent-addr              -- address of object generating  
CLNPAddress,        -- trap
  generic-trap            -- generic trap type  
  INTEGER {  
    coldStart(0),  
    warmStart(1),  
    linkDown(2),  
    linkUp(3),  
    authenticationFailure(4),  
    egpNeighborLoss(5),  
    enterpriseSpecific(6)  
  },
specific-trap           -- specific code, present even  
  INTEGER,            -- if generic-trap is not  
  -- enterpriseSpecific
  time-stamp             -- time elapsed between the last  
    TimeTicks,        -- (re)initialization of the  
    -- network entity and the  
    -- generation of the trap
variable-bindings       -- "interesting" information  
  VarBindList
}

END

5. Acknowledgements

The predecessor of this document (RFC 1161) was produced by the SNMP  
Working Group, and subsequently modified by the editor to reflect  
operational experience gained since the original publication.

6. References

  Network Management Protocol (SNMP)", RFC 1157, SNMP Research,  
  Performance Systems International, Performance Systems

Rose                                                   [Page 7]


7. Security Considerations

Security issues are not discussed in this memo.

8. Author’s Address

Marshall T. Rose
Dover Beach Consulting, Inc.
420 Whisman Court
Mountain View, CA 94043-2112

Phone: (415) 968-1052
Email: mrose@dbc.mtview.ca.us
X.500: mrose, dbc, us