Network Working Group   Thomas D. Nadeau
Internet-Draft   Cisco Systems, Inc.
Expires: May 2003   Joan Cucchiara
                                                            Consultant
                                                       Cheenu Srinivasan
                                                     Parama Networks, Inc.
                                                       Arun Viswanathan
                                                     Force10 Networks, Inc.
                                                         Hans Sjostrand
                                                            ipUnplugged

                                                          November 2002

Definitions of Textual Conventions for Multiprotocol Label Switching (MPLS) Management

<draft-ietf-mpls-tc-mib-05.txt>

Status of this Memo

Copyright Notice

Abstract

   This memo describes Textual Conventions for use in definitions of
   management information for Multiprotocol Label Switching (MPLS)
   networks.

Table of Contents

1. Introduction

This document defines a MIB which contains Textual Conventions for
use in definitions of management information for Multi-Protocol Label
Switching (MPLS) networks.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in RFC 2119 [21].

For an introduction to the concepts of MPLS, see [RFC3031].

2. The SNMP Management Framework

The SNMP Management Framework presently consists of five major
components:

o   An overall architecture, described in RFC 2571 [RFC2571].

o   Mechanisms for describing and naming objects and events for the
     purpose of management. The first version of this Structure of
     Management Information (SMI) is called SMIv1 and described in
     STD 16, RFC 1155 [RFC1155], STD 16, RFC 1212 [RFC1212] and RFC
     1215 [RFC1215]. The second version, called SMIv2, is described
     in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and
     STD 58, RFC 2580 [RFC2580].

o   Message protocols for transferring management information. The
     first version of the SNMP message protocol is called SNMPv1 and
     described in STD 15, RFC 1157 [RFC1157]. A second version of
     the SNMP message protocol, which is not an Internet standards
     track protocol, is called SNMPv2c and described in RFC 1901
     [RFC1901] and RFC 1906 [RFC1906]. The third version of the
     message protocol is called SNMPv3 and described in RFC 1906
     [RFC1906], RFC 2572 [RFC2572] and RFC 2574 [RFC2574].

o   Protocol operations for accessing management information. The
     first set of protocol operations and associated PDU formats is
     described in STD 15, RFC 1157 [RFC1157]. A second set of
     protocol operations and associated PDU formats is described in
     RFC 1905 [RFC1905].

o   A set of fundamental applications described in RFC 2573
     [RFC2573] and the view-based access control mechanism described
     in RFC 2575 [RFC2575].

A more detailed introduction to the current SNMP Management Framework
can be found in RFC 2570 [RFC2570].

Managed objects are accessed via a virtual information store, termed
the Management Information Base or MIB. Objects in the MIB are
defined using the mechanisms defined in the SMI.

This memo specifies a MIB module that is compliant to the SMIv2. A
MIB conforming to the SMIv1 can be produced through the appropriate
translations. The resulting translated MIB must be semantically
equivalent, except where objects or events are omitted because no
translation is possible. Some machine readable information in SMIv2
will be converted into textual descriptions in SMIv1 during the
translation process. However, this loss of machine readable
information is not considered to change the semantics of the MIB.

3. **MPLS Textual Conventions MIB Definitions**

   MPLS-TC-MIB DEFINITIONS ::= BEGIN

        IMPORTS

            MODULE-IDENTITY, Unsigned32, Integer32, transmission
                FROM SNMPv2-SMI

            TEXTUAL-CONVENTION
                FROM SNMPv2-TC;

        mplsTCMIB MODULE-IDENTITY
            LAST-UPDATED "200211031200Z" -- 3 Nov 2002 12:00:00 GMT
            ORGANIZATION
                "IETF Multiprotocol Label Switching (MPLS) Working
                 Group."
            CONTACT-INFO
                "   Thomas D. Nadeau
                        Cisco Systems, Inc.
                        tnadeau@cisco.com

                        Joan Cucchiara
                        Consultant
                        jcucchiara@mindspring.com

                        Cheenu Srinivasan
                        Parama Networks, Inc.
                        cheenu@paramanet.com

                        Arun Viswanathan
                        Force10 Networks, Inc.
                        arun@force10networks.com

                        Hans Sjostrand

ipUnplugged

                    hans@ipunplugged.com

            Email comments to the MPLS WG Mailing List at
            mpls@uu.net."
        DESCRIPTION
            "This MIB module defines Textual Conventions
             for use in definitions of management
             information for Multi-Protocol Label Switching
             (MPLS) networks."

        REVISION "200211031200Z" -- 3 Nov 2002 12:00:00 GMT
        DESCRIPTION
            "Initial version published as part of RFC XXXX."
        ::= { mplsMIB 1 }

    -- This object identifier needs to be assigned by IANA.
    -- Since mpls has been assigned an ifType of 166 we recommend
    -- that this OID be 166 as well.

    mplsMIB OBJECT IDENTIFIER
        ::= { transmission XXX }

    -- Textual Conventions are in alphabetical order.

    MplsAtmVcIdentifier ::= TEXTUAL-CONVENTION
        DISPLAY-HINT "d"
        STATUS   current
        DESCRIPTION
            "A Label Switching Router (LSR) that
             creates LDP sessions on ATM interfaces
             uses the VCI or VPI/VCI field to hold the
             LDP Label.

             VCI values MUST NOT be in the 0-31 range.
             The values 0 to 31 are reserved for other uses
             by the ITU and ATM Forum. The value
             of 32 can only be used for the Control VC,
             although values greater than 32 could be
             configured for the Control VC.

             If a value from 0 to 31 is used for a VCI
             the management entity controlling the LDP
             subsystem should reject this with an
             inconsistentValue error. Also, if
             the value of 32 is used for a VC which is
             NOT the Control VC, this should
             result in an inconsistentValue error."
        REFERENCE

"[RFC3035] Davie, B., Lawerence J., McCloghrie, K.,

                    Rosen, E., Swallow G., Rekhter, Y., and
                    P. Doolan, 'MPLS using LDP and ATM VC Switching',
                    RFC 3035, January 2001."
               SYNTAX Integer32 (32..65535)

          MplsBitRate ::= TEXTUAL-CONVENTION
               DISPLAY-HINT "d"
               STATUS   current
               DESCRIPTION
                    "An estimate of bandwidth in units of 1,000 bits per
                     second. If this object reports a value of 'n' then
                     the rate of the object is somewhere in the range of
                     'n-500' to 'n+499'. For objects which do not vary
                     in bit rate, or for those where no accurate
                     estimation can be made, this object should contain
                     the nominal bit rate. A value of 0 indicates best
                     effort treatment."
               SYNTAX Integer32 (0|500..2147483647)

          MplsBurstSize ::= TEXTUAL-CONVENTION
               DISPLAY-HINT "d"
               STATUS   current
               DESCRIPTION
                    "The number of octets of MPLS data that the stream
                     may send back-to-back without concern for policing.
                     The value of zero indicates that an implementation
                     does not support Burst Size."
               SYNTAX Unsigned32 (0..4294967295)

          MplsExtendedTunnelId ::= TEXTUAL-CONVENTION
               STATUS   current
               DESCRIPTION
                    "A unique identifier for an MPLS Tunnel. This may
                     represent an IPv4 address of the ingress or egress
                     LSR for the tunnel. This value is derived from the
                     Extended Tunnel Id in RSVP or the Ingress Router ID
                     for CR-LDP."
               REFERENCE
                    "[RFC3209] Awduche, D., et al., 'RSVP-TE: Extensions
                     to RSVP for LSP Tunnels', RFC 3209, December 2001.

                     [RFC3212] Jamoussi, B., et al., 'Constraint-Based
                     LSP Setup using LDP', RFC 3212, January 2002."
               SYNTAX Unsigned32

          MplsOwner ::= TEXTUAL-CONVENTION
               STATUS   current
               DESCRIPTION

"The entity that originally created the object in
question. The values of this enumeration are

defined as follows:

other(1) - This is used when an entity which has not
been enumerated in this textual convention but
which is known by the agent.

snmp(2) - The Simple Network Management Protocol was
used to configure this object initially.

ldp(3 - The Label Distribution Protocol was used to
configure this object initially.

rsvp(4) - The Resource Reservation Protocol was used
to configure this object initially.

crldp(5) - The Constraint-Based Label Distribution
Protocol was used to configure this object
initially.

policyAgent(6) - A policy agent (perhaps in
combination with one of the above protocols) was
used to configure this object initially.

unknown(7) - the agent cannot discern which
component created the object.

An object created by the ldp(3), rsvp(4), crldp(5)
or policyAgent(6) MAY be modified through operator
intervention using other(1) or snmp(2). In
particular, operators may bring rows in and
out of service or modify their values.
In all other respects, the MplsOwner is
the only source allowed to modify the status of
the object.

Agents receiving requests which violate these
guidelines MUST return an inconsistentValue(12)
error."
            SYNTAX INTEGER {
                    other(1),
                    snmp(2),
                    ldp(3),
                    rsvp(4),
                    crldp(5),
                    policyAgent(6),
                    unknown (7)
                }

        MplsLSPID ::= TEXTUAL-CONVENTION

STATUS   current

                      DESCRIPTION
                            "A unique identifier within an MPLS network that is
                             assigned to each LSP. This is assigned at the head
                             end of the LSP and can be used by all LSRs
                             to identify this LSP. This value is piggybacked by
                             the signaling protocol when this LSP is signaled
                             within the network. This identifier can then be
                             used at each LSR to identify which labels are being
                             swapped to other labels for this LSP. This object
                             can also be used to disambiguate LSPs that
                             share the same RSVP sessions between the same
                             source and destination.

                             For LSPs established using CR-LDP, the LSPID is
                             composed of the ingress LSR Router ID (or any of
                             its own IPv4 addresses) and a locally unique
                             CR-LSP ID to that LSR. The first two bytes carry
                             the CR-LSPID, and the remaining 4 bytes carry
                             the Router ID. The LSPID is useful in network
                             management, in CR-LSP repair, and in using
                             an already established CR-LSP as a hop in an ER-TLV.

                             For LSPs signaled using RSVP-TE, the LSP ID is
                             defined as a 16-bit (2 byte) identifier used
                             in the SENDER_TEMPLATE and the FILTER_SPEC
                             that can be changed to allow a sender to
                             share resources with itself. The length of this
                             object should only be 2 or 6 bytes. If the length
                             of this octet string is 2 bytes, then it must
                             identify an RSVP-TE LSPID, or it is 6 bytes,
                             it must contain a CR-LDP LSPID."
                      REFERENCE
                            "See [RFC3209] for RSVP-TE LSPID and [RFC3212] for
                             LSPID in CR-LDP."
                      SYNTAX OCTET STRING (SIZE (2|6))

                   MplsLabel ::= TEXTUAL-CONVENTION
                      STATUS   current
                      DESCRIPTION
                            "This value represents an MPLS label as defined in
                             [RFC3031], [RFC3032], [RFC3034], [RFC3035] and
                             [CCAMP-ARCH].

                             The label contents are specific to the label being
                             represented, such as:

                             * The label carried in an MPLS shim header
                               (for LDP this is the Generic Label) is a 20-bit

number represented by 4 octets. Bits 0-19 contain
a label or a reserved label value. Bits 20-31

MUST be zero.

The following is quoted directly from [RFC3032].
There are several reserved label values:

   i. A value of 0 represents the
     'IPv4 Explicit NULL Label'. This label
     value is only legal at the bottom of the
     label stack. It indicates that the label
     stack must be popped, and the forwarding
     of the packet must then be based on the
     IPv4 header.

  ii. A value of 1 represents the
     'Router Alert Label'. This label value is
     legal anywhere in the label stack except at
     the bottom. When a received packet
     contains this label value at the top of
     the label stack, it is delivered to a
     local software module for processing.
     The actual forwarding of the packet
     is determined by the label beneath it
     in the stack. However, if the packet is
     forwarded further, the Router Alert Label
     should be pushed back onto the label stack
     before forwarding. The use of this label
     is analogous to the use of the
     'Router Alert Option' in IP packets [5]
     [Reference to RFC2113]. Since this label
     cannot occur at the bottom of the stack,
     it is not associated with a
     particular network layer protocol.

 iii. A value of 2 represents the
     'IPv6 Explicit NULL Label'. This label
     value is only legal at the bottom of the
     label stack. It indicates that the label
     stack must be popped, and the forwarding
     of the packet must then be based on the
     IPv6 header.

  iv. A value of 3 represents the
     'Implicit NULL Label'.
     This is a label that an LSR may assign and
     distribute, but which never actually
     appears in the encapsulation. When an
     LSR would otherwise replace the label
     at the top of the stack with a new label,

but the new label is 'Implicit NULL',
the LSR will pop the stack instead of

                              doing the replacement. Although
                              this value may never appear in the
                              encapsulation, it needs to be specified in
                              the Label Distribution Protocol, so a value
                              is reserved.

                        v. Values 4-15 are reserved.

                  * The frame relay label can be either 10-bits or
                    23-bits depending on the DLCI field size and the
                    upper 22-bits or upper 9-bits must be zero,
                    respectively.

                  * For an ATM label the lower 16-bits represents the
                    VCI, the next 12-bits represents the VPI and the
                    remaining bits MUST be zero.

                  * The Generalized-MPLS (GMPLS) label contains a
                    value greater than 2^24-1 and used in GMPLS
                    as defined in [CCAMP-ARCH]."

             REFERENCE
                  "[RFC3031] Multiprotocol Label Switching
                  Architecture, Rosen et al., RFC 3031, August 1999.

                  [RFC3032] MPLS Label Stack Encoding, Rosen et al.,
                  RFC 3032, January 2001.

                  [RFC3034] Use of Label Switching on Frame Relay
                  Networks, Conta et al., RFC 3034, January 2001.

                  [RFC3035] MPLS using LDP and ATM VC Switching,
                  Davie et al., RFC 3035, January 2001.

                  [CCAMP-ARCH] Generalized Multi-Protocol Label
                  Switching (GMPLS) Architecture, Mannie (Editor),
                  draft-ietf-ccamp-gmpls-architecture-02.txt,
                  March 2002."
             SYNTAX Unsigned32 (0..4294967295)

        MplsLabelDistributionMethod ::= TEXTUAL-CONVENTION
             STATUS current
             DESCRIPTION
                  "The label distribution method which is also called
                  the label advertisement mode (see LDP Specification).
                  Each interface on an LSR is configured to operate
                  in either Downstream Unsolicited or Downstream
                  on Demand."
             REFERENCE

"[RFC3031] Multiprotocol Label Switching

                    Architecture, Rosen et al., RFC 3031, August 1999.

                    [RFC3036] LDP Specification, Andersson, L., et. al.,
                    RFC 3036, Section 2.6.3., January 2001."
              SYNTAX INTEGER {
                      downstreamOnDemand(1),
                      downstreamUnsolicited(2)
                   }

          MplsLspType ::= TEXTUAL-CONVENTION
             STATUS current
             DESCRIPTION
                  "Types types of Label Switch Paths (LSPs)
                   on an Label Switching Router (LSR) are:

                      unknown(1)   -- if the LSP is not known
                                             to be one of the following.

                      terminatingLsp(2) -- if the LSP terminates
                                             on the LSR, then this
                                             is an ingressing LSP
                                             which ends on the LSR,

                      originatingLsp(3) -- if the LSP originates
                                             from the LSR, then this
                                             is an egressing LSP which is
                                             the head-end of the LSP,

                   crossConnectingLsp(4) -- if the LSP ingresses
                                             and egresses on the LSR,
                                             then it is cross-connecting
                                             on that LSR."
              SYNTAX INTEGER {
                      unknown(1),
                      terminatingLsp(2),
                      originatingLsp(3),
                      crossConnectingLsp(4)
                   }

          MplsLsrIndex ::= TEXTUAL-CONVENTION
             STATUS current
             DESCRIPTION
                  "Represents a generic index used throughout the
                   MPLS-LSR-MIB as a general index in the
                   mplsInSegmentTable, mplsOutSegmentTable
                   and mplsXCTable."
              SYNTAX OCTET STRING (SIZE(1..34))

```
        MplsRetentionMode ::= TEXTUAL-CONVENTION
```

```
        STATUS  current
        DESCRIPTION
            "The label retention mode which specifies whether
             an LSR maintains a label binding for a FEC learned
             from a neighbor that is not its next hop for the
             FEC.

             If the value is conservative(1) then advertised
             label mappings are retained only if they will be
             used to forward packets, i.e. if label came from
             a valid next hop.

             If the value is liberal(2) then all advertised label
             mappings are retained whether they are from a
             valid next hop or not."
        REFERENCE
            "[RFC3031] Multiprotocol Label Switching
             Architecture, Rosen et al., RFC 3031, August 1999.

             [RFC3036] LDP Specification, Andersson, L., et. al.,
             RFC 3036, Section 2.6.2., January 2001."
        SYNTAX INTEGER {
                conservative(1),
                liberal(2)
             }

    MplsLdpIdentifier ::= TEXTUAL-CONVENTION
       STATUS      current
       DESCRIPTION
            "The LDP identifier is a six octet quantity which is
             used to identify an Label Switching Router (LSR)
             label space.

             The first four octets identify the LSR and must be
             a globally unique value, such as a 32-bit router ID
             assigned to the LSR, and the last two octets
             identify a specific label space within the LSR."
       SYNTAX  OCTET STRING (SIZE (6))

    MplsLdpLabelType ::= TEXTUAL-CONVENTION
       STATUS      current
       DESCRIPTION
            "The Layer 2 label types which are defined for MPLS
             LDP and/or CR-LDP are generic(1), atm(2), or
             frameRelay(3)."
       SYNTAX  INTEGER {
                generic(1),
                atm(2),
```

```
                    frameRelay(3)
         }
```

```
MplsLsrIdentifier ::= TEXTUAL-CONVENTION
   STATUS        current
   DESCRIPTION
        "The Label Switching Router (LSR) identifier is the
         first 4 bytes of the Label Distribution Protocol
         (LDP) identifier."
   SYNTAX  OCTET STRING (SIZE (4))

MplsPathIndex ::= TEXTUAL-CONVENTION
   STATUS        current
   DESCRIPTION
        "A unique value to index (by Path number) an entry
         in a table."
   SYNTAX  Unsigned32(1..4294967295)

MplsPathIndexOrZero ::= TEXTUAL-CONVENTION
   STATUS current
   DESCRIPTION
        "A unique identifier used to identify a specific path
         used by a tunnel. A value of 0 (zero) means that
         no path is in use."
   SYNTAX  Unsigned32

MplsTunnelAffinity ::= TEXTUAL-CONVENTION
   STATUS        current
   DESCRIPTION
        "Describes the configured 32-bit Include-any,
         include-all, or exclude-all constraint for
         constraint-based link selection."
   REFERENCE
        "See section 4.7.4 in [RFC3209]."
   SYNTAX  Unsigned32

MplsTunnelIndex ::= TEXTUAL-CONVENTION
   STATUS        current
   DESCRIPTION
        "A unique index into mplsTunnelTable.
         For tunnels signaled using RSVP, this value
         should correspond to the RSVP destination
         port used for the RSVP-TE session."
   SYNTAX  Integer32 (1..65535)

MplsTunnelInstanceIndex ::= TEXTUAL-CONVENTION
   STATUS        current
   DESCRIPTION
        "Instance index into mplsTunnelTable. The
         tunnel entry with instance index 0 should
         refer to the configured tunnel interface
```

(if one exists), and values greater an 0
should be used to indicate signaled (or backup)

                    tunnel LSP instances. For tunnel LSPs signaled using
                    RSVP, this value should correspond to the
                    RSVP source port used for the RSVP-TE session."
            SYNTAX Unsigned32 (0..65535)

        END

4. **References**

[RFC3212]   Jamoussi, B., (editor), et. al. "Constraint-Based LSP Setup
             using LDP", RFC 3212, January 2002.

[RFC3209]   Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V.,
             Swallow, G., "RSVP-TE: Extensions to RSVP for LSP Tunnels",
             RFC 3209, December 2001.

[RFC3031]   Rosen, E., Viswananthan, A., and R. Callon, "Multiprotocol
             Label Switching Architecture", RFC 3031, January 2001.

[RFC3032]   Rosen, E., Rekhter, Y., Tappan, D., Farinacci, D.,
             Federokow, G., Li, T., and A. Conta, "MPLS Label Stack
             Encoding", RFC 3032, January 2001.

[RFC3034]   Conta, A., Doolan, P., and A. Malis, "Use of Label Switching
             on Frame Relay Networks Specification", RFC 3034, January
             2001.

[RFC3035]   Davie, B., Lawrence, J., McCloghrie, K., Rosen, E., Swallow,
             G., Rekhter, Y., and P. Doolan, "MPLS using LDP and ATM VC
             Switching", RFC 3035, January 2001.

[RFC3036]   Andersson, L., Doolan, P., Feldman, N., Fredette, A., and B.
             Thomas, "LDP Specification", RFC 3036, January 2001.

[RFC2571]   Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture
             for Describing SNMP Management Frameworks", RFC 2571, April
             1999.

[RFC1155]   Rose, M., and K. McCloghrie, "Structure and Identification
             of Management Information for TCP/IP-based Internets", STD
             16, RFC 1155, May 1990.

[RFC1212]   Rose, M., and K. McCloghrie, "Concise MIB Definitions", STD
             16, RFC 1212, March 1991.

[RFC1215]   M. Rose, "A Convention for Defining Traps for use with the
             SNMP", RFC 1215, March 1991.

[RFC2578]   McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,
             Rose, M., and S. Waldbusser, "Structure of Management
             Information Version 2 (SMIv2)", STD 58, RFC 2578, April
             1999.

[RFC2579]   McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,
             Rose, M., and S. Waldbusser, "Textual Conventions for
             SMIv2", STD 58, RFC 2579, April 1999.

[RFC2580]   McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,
            Rose, M., and S. Waldbusser, "Conformance Statements for
            SMIv2", STD 58, RFC 2580, April 1999.

[RFC1157]   Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple
            Network Management Protocol", STD 15, RFC 1157, May 1990.

[RFC1901]   Case, J., McCloghrie, K., Rose, M., and S. Waldbusser,
            "Introduction to Community-based SNMPv2", RFC 1901, January
            1996.

[RFC1906]   Case, J., McCloghrie, K., Rose, M., and S. Waldbusser,
            "Transport Mappings for Version 2 of the Simple Network
            Management Protocol (SNMPv2)", RFC 1906, January 1996.

[RFC2572]   Case, J., Harrington D., Presuhn R., and B. Wijnen, "Message
            Processing and Dispatching for the Simple Network Management
            Protocol (SNMP)", RFC 2572, April 1999.

[RFC2574]   Blumenthal, U., and B. Wijnen, "User-based Security Model
            (USM) for version 3 of the Simple Network Management
            Protocol (SNMPv3)", RFC 2574, April 1999.

[RFC1905]   Case, J., McCloghrie, K., Rose, M., and S. Waldbusser,
            "Protocol Operations for Version 2 of the Simple Network
            Management Protocol (SNMPv2)", RFC 1905, January 1996.

[RFC2573]   Levi, D., Meyer, P., and B. Stewart, "SNMPv3 Applications",
            RFC 2573, April 1999.

[RFC2575]   Wijnen, B., Presuhn, R., and K. McCloghrie, "View-based
            Access Control Model (VACM) for the Simple Network
            Management Protocol (SNMP)", RFC 2575, April 1999.

[RFC2570]   Case, J., Mundy, R., Partain, D., and B. Stewart,
            "Introduction to Version 3 of the Internet-standard Network
            Management Framework", RFC 2570, April 1999.

5. Security Considerations

   This module does not define any management objects. Instead, it
   defines a set of textual conventions which may be used by other MPLS
   MIB modules to define management objects.

   Meaningful security considerations can only be written in the MIB
   modules that define management objects. Therefore, this document has

no impact on the security of the Internet.

[6](#). Authors' Addresses

          Thomas D. Nadeau
          Cisco Systems, Inc.
          250 Apollo Drive
          Chelmsford, MA 01824
          Phone: +1-978-244-3051
          Email: tnadeau@cisco.com

          Joan Cucchiara
          Consultant
          PO Box 1010
          Concord, MA
          Phone: +1-508-303-8200 x302
          Email: jcucchiara@mindspring.com

          Cheenu Srinivasan
          Parama Networks, Inc.
          1030 Broad Street
          Shrewsbury, NJ 07702
          Phone: +1-732-544-9120 x731
          Email: cheenu@paramanet.com

          Arun Viswanathan
          Force10 Networks, Inc.
          1440 McCarthy Blvd
          Milpitas, CA 95035
          Phone: +1-408-571-3516
          Email: arun@force10networks.com

          Hans Sjostrand
          ipUnplugged
          P.O. Box 101 60
          S-121 28 Stockholm, Sweden
          Phone: +46-8-725-5930
          Email: hans@ipunplugged.com

[7](#). **Full Copyright Statement**

kind, provided that the above copyright notice and this paragraph are

included on all such copies and derivative works. However, this
document itself may not be modified in any way, such as by removing
the copyright notice or references to the Internet Society or other
Internet organizations, except as needed for the purpose of
developing Internet standards in which case the procedures for
copyrights defined in the Internet Standards process must be
followed, or as required to translate it into languages other than
English.

The limited permissions granted above are perpetual and will not be
revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an
"AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING
TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION
HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF
MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.