

## Textual Conventions for Transport Addresses

### Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

### Copyright Notice

Copyright (C) The Internet Society (2002). All Rights Reserved.

### Abstract

This document introduces a Management Information Base (MIB) module that defines textual conventions to represent commonly used transport-layer addressing information. The definitions are compatible with the concept of TAddress/TDomain pairs introduced by the Structure of Management Information version 2 (SMIv2) and support the Internet transport protocols over IPv4 and IPv6.

### Table of Contents

1.	Introduction . . . . .	2
2.	The Internet-Standard Management Framework . . . . .	2
3.	Overview . . . . .	3
3.1	Relationship to Other MIBs . . . . .	4
3.1.1	SNMPv2-TC (TAddress, TDomain) . . . . .	4
3.1.2	SNMPv2-TM . . . . .	4
3.1.3	INET-ADDRESS-MIB (InetAddressType, InetAddress) . . . . .	5
4.	Definitions . . . . .	5
5.	Examples . . . . .	14
6.	Security Considerations . . . . .	15
7.	Acknowledgments . . . . .	15
8.	Intellectual Property Notice . . . . .	15
	Normative References . . . . .	16
	Informative References . . . . .	16
	Authors' Addresses . . . . .	17
	Full Copyright Statement . . . . .	18

## 1. Introduction

Several MIB modules need to represent transport-layer addresses in a generic way. Typical examples are MIBs for application protocols that can operate over several different transports or application management MIBs that need to model generic communication endpoints.

The SMIV2 in STD 58, RFC 2579 [RFC2579] defines the textual conventions TDomain and TAddress to represent generic transport layer endpoints. A generic TAddress value is interpreted in a given transport domain which is identified by a TDomain value. The TDomain is an object identifier which allows MIB authors to extend the set of supported transport domains by providing suitable definitions in standardized or enterprise specific MIB modules.

An initial set of TDomain values and concrete TAddress formats has been standardized in STD 62, RFC 3417 [RFC3417]. These definitions are however mixed up with SNMP semantics. Furthermore, definitions for Internet transport protocols over IPv4 and IPv6 are missing.

The purpose of this memo is to introduce a set of well-known textual conventions to represent commonly used transport-layer addressing information which is compatible with the original TDomain and TAddress approach and which includes definitions for additional Internet transport protocols over IPv4 and IPv6. This memo also introduces a new textual convention which enumerates the well-known transport domains since such an enumeration provides in many cases sufficient flexibility and is more efficient compared to object identifiers.

The key words "MUST", "MUST NOT", "SHOULD", "SHOULD NOT" and "MAY" in this document are to be interpreted as described in BCP 14, RFC 2119 [RFC2119].

## 2. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of RFC 3410 [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIV2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

### 3. Overview

This MIB module contains definitions for commonly used transport layer addressing information. In particular, it provides the following definitions:

1. Textual conventions for generic transport addresses (TransportAddress) and generic transport domains (TransportDomain).
2. Object identifier registrations for well-known transport domains.
3. An enumeration of the well-known transport domains, called a transport address type (TransportAddressType).
4. A set of textual conventions for the address formats used by well-known transport domains. The DISPLAY-HINTs are aligned with the formats used in URIs [RFC2396], [RFC3291].

The textual conventions for well-known transport domains support scoped Internet addresses. The scope of an Internet address is a topological span within which the address may be used as a unique identifier for an interface or set of interfaces. A scope zone, or simply a zone, is a concrete connected region of topology of a given scope. Note that a zone is a particular instance of a topological region, whereas a scope is the size of a topological region [SCOPED]. Since Internet addresses on devices that connect multiple zones are not necessarily unique, an additional zone index is needed on these devices to select an interface. The textual conventions TransportAddressIPv4z and TransportAddressIPv6z are provided to support Internet transport addresses that include a zone index. In order to support arbitrary combinations of scoped Internet transport addresses, MIB authors SHOULD use a separate TransportDomain or TransportAddressType objects for each TransportAddress object.

There are two different ways how new transport domains and textual conventions for the address formats used by those new transport domains can be defined.

1. The MIB module contained in this memo can be updated and new constants for the TransportDomain and the TransportAddressType enumeration can be assigned.
2. Other MIB modules may define additional transport domains and associated textual conventions. Such an extension can not update the TransportAddressType enumeration.

It is therefore a MIB designers choice whether he uses (a) a more compact TransportAddressType object with limited extensibility or (b) a more verbose TransportDomain object which allows arbitrary extensions in other MIB modules.

The MIB module contained in this memo does NOT define the transport mappings of any particular protocol. Rather, it defines a set of common identifiers and textual conventions that are intended to be used within various transport mappings documents.

### 3.1 Relationship to Other MIBs

This section discusses how the definitions provided by the MIB module contained in this memo relate to definitions in other MIB modules.

#### 3.1.1 SNMPv2-TC (TAddress, TDomain)

The SNMPv2-TC MIB module [RFC2579] defines the textual conventions TAddress and TDomain to represent generic transport addresses.

A TAddress is an octet string with a size between 1 and 255 octets. Experience has shown that there is sometimes a need to represent unknown transport addresses. The MIB module contained in this memo therefore introduces a new textual convention TransportAddress which is an octet string with a size between 0 and 255 octets and otherwise identical semantics. In other words, the sub-type TransportAddress (SIZE (1..255)) is identical with the TAddress defined in the SNMPv2-TC MIB module [RFC2579].

This MIB module also introduces a new textual convention TransportDomain which is compatible with the TDomain definition so that a complete set of definitions is contained in a single MIB module. New MIB modules SHOULD use the generic TransportDomain, TransportAddressType and TransportAddress definitions defined in this memo. Existing MIB modules may be updated to use the definitions provided in this memo by replacing TDomain with TransportDomain and TAddress with TransportAddress (SIZE (1..255)).

#### 3.1.2 SNMPv2-TM

The transport domain values defined in the SNMPv2-TM MIB module [RFC3417] all contain "snmp" as the prefix in their name and are registered under 'snmpDomains' (from RFC 2578 [RFC2578]). They were originally intended to describe SNMP transport domains only - but they were later also used for non-SNMP transport endpoints. These definitions are also incomplete since new transport address domains are needed to support (at least) SNMP over UDP over IPv6.

The transport domain values defined in this memo are independent of the protocol running over the transport-layer and SHOULD be used for all transport endpoints not carrying SNMP traffic. Programs that interpret transport domain values should in addition accept the transport domain values defined in the SNMPv2-TM MIB module in order to provide interoperability with existing implementations that use the SNMP specific transport domain values.

Transport endpoints which carry SNMP traffic SHOULD continue to use the definitions from the SNMPv2-TM MIB module where applicable. They SHOULD use the transport domain values defined in this memo for SNMP transports not defined in the SNMPv2-TM MIB module, such as SNMP over UDP over IPv6. Programs that interpret transport domain values should in addition accept all the transport domain values defined in this memo in order to provide interoperability in cases where it is not possible or desirable to distinguish the protocols running over a transport endpoint.

### 3.1.3 INET-ADDRESS-MIB (InetAddressType, InetAddress)

The INET-ADDRESS-MIB MIB module [RFC3291] defines the textual conventions InetAddressType and InetAddress to represent Internet network layer endpoints. Some MIB modules use these textual conventions in conjunction with the InetPortNumber textual convention to represent Internet transport-layer endpoints. This approach is fine as long as a MIB models protocols or applications that are specific to the Internet suite of transport protocols. For protocols or applications that can potentially use other transport protocols, the use of the definitions contained in this memo is more appropriate.

## 4. Definitions

```
TRANSPORT-ADDRESS-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    MODULE-IDENTITY, OBJECT-IDENTITY, mib-2          FROM SNMPv2-SMI
    TEXTUAL-CONVENTION                                FROM SNMPv2-TC;
```

```
transportAddressMIB MODULE-IDENTITY
```

```
    LAST-UPDATED "200211010000Z"
```

```
    ORGANIZATION
```

```
        "IETF Operations and Management Area"
```

```
    CONTACT-INFO
```

```
        "Juergen Schoenwaelder (Editor)"
```

```
        TU Braunschweig
```

```
        Bueltenweg 74/75
```

```
        38106 Braunschweig, Germany
```

Phone: +49 531 391-3289  
 EMail: schoenw@ibr.cs.tu-bs.de

Send comments to <mibs@ops.ietf.org>."

DESCRIPTION

"This MIB module provides commonly used transport address definitions.

Copyright (C) The Internet Society (2002). This version of this MIB module is part of RFC 3419; see the RFC itself for full legal notices."

-- Revision log

REVISION        "200211010000Z"

DESCRIPTION

"Initial version, published as RFC 3419."

::= { mib-2 100 }

transportDomains OBJECT IDENTIFIER ::= { transportAddressMIB 1 }

transportDomainUdpIpv4 OBJECT-IDENTITY

STATUS            current

DESCRIPTION

"The UDP over IPv4 transport domain. The corresponding transport address is of type TransportAddressIPv4 for global IPv4 addresses."

::= { transportDomains 1 }

transportDomainUdpIpv6 OBJECT-IDENTITY

STATUS            current

DESCRIPTION

"The UDP over IPv6 transport domain. The corresponding transport address is of type TransportAddressIPv6 for global IPv6 addresses."

::= { transportDomains 2 }

transportDomainUdpIpv4z OBJECT-IDENTITY

STATUS            current

DESCRIPTION

"The UDP over IPv4 transport domain. The corresponding transport address is of type TransportAddressIPv4z for scoped IPv4 addresses with a zone index."

::= { transportDomains 3 }

transportDomainUdpIpv6z OBJECT-IDENTITY

STATUS            current

## DESCRIPTION

"The UDP over IPv6 transport domain. The corresponding transport address is of type TransportAddressIPv6z for scoped IPv6 addresses with a zone index."

::= { transportDomains 4 }

## transportDomainTcpIpv4 OBJECT-IDENTITY

STATUS current

## DESCRIPTION

"The TCP over IPv4 transport domain. The corresponding transport address is of type TransportAddressIPv4 for global IPv4 addresses."

::= { transportDomains 5 }

## transportDomainTcpIpv6 OBJECT-IDENTITY

STATUS current

## DESCRIPTION

"The TCP over IPv6 transport domain. The corresponding transport address is of type TransportAddressIPv6 for global IPv6 addresses."

::= { transportDomains 6 }

## transportDomainTcpIpv4z OBJECT-IDENTITY

STATUS current

## DESCRIPTION

"The TCP over IPv4 transport domain. The corresponding transport address is of type TransportAddressIPv4z for scoped IPv4 addresses with a zone index."

::= { transportDomains 7 }

## transportDomainTcpIpv6z OBJECT-IDENTITY

STATUS current

## DESCRIPTION

"The TCP over IPv6 transport domain. The corresponding transport address is of type TransportAddressIPv6z for scoped IPv6 addresses with a zone index."

::= { transportDomains 8 }

## transportDomainSctpIpv4 OBJECT-IDENTITY

STATUS current

## DESCRIPTION

"The SCTP over IPv4 transport domain. The corresponding transport address is of type TransportAddressIPv4 for global IPv4 addresses. This transport domain usually represents the primary address on multihomed SCTP endpoints."

::= { transportDomains 9 }

```

transportDomainSctpIpv6 OBJECT-IDENTITY
    STATUS          current
    DESCRIPTION
        "The SCTP over IPv6 transport domain. The corresponding
        transport address is of type TransportAddressIPv6 for
        global IPv6 addresses. This transport domain usually
        represents the primary address on multihomed SCTP
        endpoints."
    ::= { transportDomains 10 }

transportDomainSctpIpv4z OBJECT-IDENTITY
    STATUS          current
    DESCRIPTION
        "The SCTP over IPv4 transport domain. The corresponding
        transport address is of type TransportAddressIPv4z for
        scoped IPv4 addresses with a zone index. This transport
        domain usually represents the primary address on
        multihomed SCTP endpoints."
    ::= { transportDomains 11 }

transportDomainSctpIpv6z OBJECT-IDENTITY
    STATUS          current
    DESCRIPTION
        "The SCTP over IPv6 transport domain. The corresponding
        transport address is of type TransportAddressIPv6z for
        scoped IPv6 addresses with a zone index. This transport
        domain usually represents the primary address on
        multihomed SCTP endpoints."
    ::= { transportDomains 12 }

transportDomainLocal OBJECT-IDENTITY
    STATUS          current
    DESCRIPTION
        "The Posix Local IPC transport domain. The corresponding
        transport address is of type TransportAddressLocal.

        The Posix Local IPC transport domain incorporates the
        well-known UNIX domain sockets."
    ::= { transportDomains 13 }

transportDomainUdpDns OBJECT-IDENTITY
    STATUS          current
    DESCRIPTION
        "The UDP transport domain using fully qualified domain
        names. The corresponding transport address is of type
        TransportAddressDns."
    ::= { transportDomains 14 }

```



transportDomainTcpDns OBJECT-IDENTITY

STATUS            current

DESCRIPTION

"The TCP transport domain using fully qualified domain names. The corresponding transport address is of type TransportAddressDns."

::= { transportDomains 15 }

transportDomainSctpDns OBJECT-IDENTITY

STATUS            current

DESCRIPTION

"The SCTP transport domain using fully qualified domain names. The corresponding transport address is of type TransportAddressDns."

::= { transportDomains 16 }

TransportDomain ::= TEXTUAL-CONVENTION

STATUS            current

DESCRIPTION

"A value that represents a transport domain.

Some possible values, such as transportDomainUdpIpv4, are defined in this module. Other possible values can be defined in other MIB modules."

SYNTAX            OBJECT IDENTIFIER

--

-- The enumerated values of the textual convention below should  
-- be identical to the last sub-identifier of the OID registered  
-- for the same domain.

--

TransportAddressType ::= TEXTUAL-CONVENTION

STATUS            current

DESCRIPTION

"A value that represents a transport domain. This is the enumerated version of the transport domain registrations in this MIB module. The enumerated values have the following meaning:

unknown(0)	unknown transport address type
udpIpv4(1)	transportDomainUdpIpv4
udpIpv6(2)	transportDomainUdpIpv6
udpIpv4z(3)	transportDomainUdpIpv4z
udpIpv6z(4)	transportDomainUdpIpv6z
tcpIpv4(5)	transportDomainTcpIpv4
tcpIpv6(6)	transportDomainTcpIpv6
tcpIpv4z(7)	transportDomainTcpIpv4z

```

tcpIpv6z(8)      transportDomainTcpIpv6z
sctpIpv4(9)      transportDomainSctpIpv4
sctpIpv6(10)     transportDomainSctpIpv6
sctpIpv4z(11)    transportDomainSctpIpv4z
sctpIpv6z(12)    transportDomainSctpIpv6z
local(13)        transportDomainLocal
udpDns(14)       transportDomainUdpDns
tcpDns(15)       transportDomainTcpDns
sctpDns(16)      transportDomainSctpDns

```

This textual convention can be used to represent transport domains in situations where a syntax of TransportDomain is unwieldy (for example, when used as an index).

The usage of this textual convention implies that additional transport domains can only be supported by updating this MIB module. This extensibility restriction does not apply for the TransportDomain textual convention which allows MIB authors to define additional transport domains independently in other MIB modules."

```

SYNTAX      INTEGER {
                unknown(0),
                udpIpv4(1),
                udpIpv6(2),
                udpIpv4z(3),
                udpIpv6z(4),
                tcpIpv4(5),
                tcpIpv6(6),
                tcpIpv4z(7),
                tcpIpv6z(8),
                sctpIpv4(9),
                sctpIpv6(10),
                sctpIpv4z(11),
                sctpIpv6z(12),
                local(13),
                udpDns(14),
                tcpDns(15),
                sctpDns(16)
            }

```

TransportAddress ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Denotes a generic transport address.

A TransportAddress value is always interpreted within the context of a TransportAddressType or TransportDomain value. Every usage of the TransportAddress textual convention MUST

specify the TransportAddressType or TransportDomain object which provides the context. Furthermore, MIB authors SHOULD define a separate TransportAddressType or TransportDomain object for each TransportAddress object. It is suggested that the TransportAddressType or TransportDomain is logically registered before the object(s) which use the TransportAddress textual convention if they appear in the same logical row.

The value of a TransportAddress object must always be consistent with the value of the associated TransportAddressType or TransportDomain object. Attempts to set a TransportAddress object to a value which is inconsistent with the associated TransportAddressType or TransportDomain must fail with an inconsistentValue error.

When this textual convention is used as a syntax of an index object, there may be issues with the limit of 128 sub-identifiers specified in SMIV2, STD 58. In this case, the OBJECT-TYPE declaration MUST include a 'SIZE' clause to limit the number of potential instance sub-identifiers."

SYNTAX OCTET STRING (SIZE (0..255))

TransportAddressIPv4 ::= TEXTUAL-CONVENTION

DISPLAY-HINT "1d.1d.1d.1d:2d"

STATUS current

DESCRIPTION

"Represents a transport address consisting of an IPv4 address and a port number (as used for example by UDP, TCP and SCTP):

octets	contents	encoding
1-4	IPv4 address	network-byte order
5-6	port number	network-byte order

This textual convention SHOULD NOT be used directly in object definitions since it restricts addresses to a specific format. However, if it is used, it MAY be used either on its own or in conjunction with TransportAddressType or TransportDomain as a pair."

SYNTAX OCTET STRING (SIZE (6))

TransportAddressIPv6 ::= TEXTUAL-CONVENTION

DISPLAY-HINT "0a[2x:2x:2x:2x:2x:2x:2x:2x]0a:2d"

STATUS current

DESCRIPTION

"Represents a transport address consisting of an IPv6 address and a port number (as used for example by UDP,

TCP and SCTP):

octets	contents	encoding
1-16	IPv6 address	network-byte order
17-18	port number	network-byte order

This textual convention SHOULD NOT be used directly in object definitions since it restricts addresses to a specific format. However, if it is used, it MAY be used either on its own or in conjunction with TransportAddressType or TransportDomain as a pair."

SYNTAX            OCTET STRING (SIZE (18))

TransportAddressIPv4z ::= TEXTUAL-CONVENTION

DISPLAY-HINT "1d.1d.1d.1d%4d:2d"

STATUS            current

DESCRIPTION

"Represents a transport address consisting of an IPv4 address, a zone index and a port number (as used for example by UDP, TCP and SCTP):

octets	contents	encoding
1-4	IPv4 address	network-byte order
5-8	zone index	network-byte order
9-10	port number	network-byte order

This textual convention SHOULD NOT be used directly in object definitions since it restricts addresses to a specific format. However, if it is used, it MAY be used either on its own or in conjunction with TransportAddressType or TransportDomain as a pair."

SYNTAX            OCTET STRING (SIZE (10))

TransportAddressIPv6z ::= TEXTUAL-CONVENTION

DISPLAY-HINT "0a[2x:2x:2x:2x:2x:2x:2x:2x%4d]0a:2d"

STATUS            current

DESCRIPTION

"Represents a transport address consisting of an IPv6 address, a zone index and a port number (as used for example by UDP, TCP and SCTP):

octets	contents	encoding
1-16	IPv6 address	network-byte order
17-20	zone index	network-byte order
21-22	port number	network-byte order

This textual convention SHOULD NOT be used directly in object definitions since it restricts addresses to a specific format.

However, if it is used, it MAY be used either on its own or in conjunction with TransportAddressType or TransportDomain as a pair."

SYNTAX OCTET STRING (SIZE (22))

TransportAddressLocal ::= TEXTUAL-CONVENTION

DISPLAY-HINT "1a"

STATUS current

DESCRIPTION

"Represents a POSIX Local IPC transport address:

octets	contents	encoding
all	POSIX Local IPC address	string

The Posix Local IPC transport domain subsumes UNIX domain sockets.

This textual convention SHOULD NOT be used directly in object definitions since it restricts addresses to a specific format. However, if it is used, it MAY be used either on its own or in conjunction with TransportAddressType or TransportDomain as a pair.

When this textual convention is used as a syntax of an index object, there may be issues with the limit of 128 sub-identifiers specified in SMIV2, STD 58. In this case, the OBJECT-TYPE declaration MUST include a 'SIZE' clause to limit the number of potential instance sub-identifiers."

REFERENCE

"Protocol Independent Interfaces (IEEE POSIX 1003.1g)"

SYNTAX OCTET STRING (SIZE (1..255))

TransportAddressDns ::= TEXTUAL-CONVENTION

DISPLAY-HINT "1a"

STATUS current

DESCRIPTION

"Represents a DNS domain name followed by a colon ':' (ASCII character 0x3A) and a port number in ASCII. The name SHOULD be fully qualified whenever possible.

Values of this textual convention are not directly useable as transport-layer addressing information, and require runtime resolution. As such, applications that write them must be prepared for handling errors if such values are not supported, or cannot be resolved (if resolution occurs at the time of the management operation).

The DESCRIPTION clause of TransportAddress objects that may

have TransportAddressDns values must fully describe how (and when) such names are to be resolved to IP addresses and vice versa.

This textual convention SHOULD NOT be used directly in object definitions since it restricts addresses to a specific format. However, if it is used, it MAY be used either on its own or in conjunction with TransportAddressType or TransportDomain as a pair.

When this textual convention is used as a syntax of an index object, there may be issues with the limit of 128 sub-identifiers specified in SMIV2, STD 58. In this case, the OBJECT-TYPE declaration MUST include a 'SIZE' clause to limit the number of potential instance sub-identifiers."

SYNTAX OCTET STRING (SIZE (1..255))

END

## 5. Examples

This section shows some examples how transport addresses are encoded and rendered using some of the transport address definitions.

Description: Unspecified IPv4 address on port 80.  
 Encoding (hex): 000000000050  
 Display: 0.0.0.0:80

Description: Global IPv4 address on port 80.  
 Encoding (hex): 86A922010050  
 Display: 134.169.34.1:80

Description: Unspecified IPv6 address on port 80.  
 Encoding (hex): 00000000000000000000000000000050  
 Display: [0:0:0:0:0:0:0:0]:80

Description: Global IPv6 address on port 80.  
 Encoding (hex): 108000000000000000080800200C417A0050  
 Display: [1080:0:0:0:8:800:200C:417A]:80

Description: Link-local IPv6 address with zone-index 42 on port 80.  
 Encoding (hex): FE8000000000000001000000002000000002A0050  
 Display: [FE80:0:0:0:1:0:0:200%42]:80

Description: Posix Local IPC address (UNIX domain).  
 Encoding (hex): 2F7661722F61676556E74782F6D6173746572  
 Display: /var/agentx/master

Description: Fully qualified domain name on port 80.  
Encoding (hex): 7777772E6578616D706C652E6E65743A3830  
Display: www.example.net:80

## 6. Security Considerations

The MIB module contained in this memo does not define any management objects. Instead, it defines a set of textual conventions which may be used by other MIB modules to define management objects.

Meaningful security considerations can only be written for MIB modules that define concrete management objects. This document has therefore no impact on the security of the Internet.

## 7. Acknowledgments

This document was produced by the Operations and Management Area "IPv6MIB" design team. The authors would like to thank Mark Ellison, Brian Haberman, Mike Heard, Glenn Mansfield Keeni, Erik Nordmark, Shawn A. Routhier, Bill Strahm, Dave Thaler and Bert Wijnen for their comments and suggestions.

## 8. Intellectual Property Notice

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

## Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [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.
- [RFC3417] Presuhn, R., Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Transport Mappings for the Simple Network Management Protocol (SNMP)", STD 62, RFC 3417, December 2002.

## Informative References

- [SCOPED] Deering, S., Haberman, B., Jinmei, T., Nordmark, E., Onoe, A. and B. Zill, "IPv6 Scoped Address Architecture", Work in Progress.
- [RFC2396] Berners-Lee, T., Fielding, R. and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998.
- [RFC2732] Hinden, R., Carpenter, B. and L. Masinter, "Format for Literal IPv6 Addresses in URL's", RFC 2732, August 1998.
- [RFC3291] Daniele, M., Haberman, B., Routhier, S. and J. Schoenwaelder, "Textual Conventions for Internet Network Addresses", RFC 3291, December 2001.
- [RFC3410] Case, J., Mundy, R., Partain, D. and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.



Authors' Addresses

Mike Daniele  
Consultant  
19 Pinewood Rd  
Hudson, NH 03051  
USA

Phone: +1 603 883-6365  
EMail: md@world.std.com

Juergen Schoenwaelder  
TU Braunschweig  
Bueltenweg 74/75  
38106 Braunschweig  
Germany

Phone: +49 531 391-3289  
EMail: schoenw@ibr.cs.tu-bs.de

## Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any 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.

## Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.

