

IPv6 MIB Revision Design Team
INTERNET-DRAFT
Expires: December 2002

Shawn A. Routhier, Editor
Wind River
June 2002

**Management Information Base
for the Internet Protocol (IP)
draft-ietf-ipv6-rfc2011-update-00.txt**

Status of this Document

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This document is a product of the IPv6 MIB Revision Design Team. Comments should be addressed to the authors, or the mailing list at ipng@sunroof.eng.sun.com.

Copyright Notice

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

Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes managed objects used for implementations of the Internet Protocol (IP) in an IP version independent manner.

Table of Contents

1. The SNMP Management Framework	2
2. Revision History	3
3. Overview	6
3.1. Multi-Stack Implementations	6
3.2. Discussion of Tables and Groups	7
3.2.1. General Objects	7
3.2.2. IPv4 Interface Table	7
3.2.3. IPv6 Interface Table	7
3.2.4. IP Statistics Table	7
3.2.5. Internet Address Prefix Table	9
3.2.6. Internet Address Table	9
3.2.7. Internet Address Translation Table	9
3.2.8. IPv6 Scope Identifier	9
3.2.9. Default Router Table	9
3.2.10. ICMP Statistics Tables	9
3.2.11. Deprecated Objects	10
3.2.12. Conformance and Compliance	10
4. Updating Implementations	10
4.1. Updating an implementation of the IPv4-only IP-MIB.	10
4.2. Updating an implementation of the IPv6-MIB	10
5. Definitions	10
6. Open Issues / To Do	76
7. Acknowledgements	77
8. References	77
9. Security Considerations	79
10. Editor's Contact Information	79
11. Authors	80
12. Acknowledgements	80
13. Full Copyright Statement	80

[1. The SNMP Management Framework](#)

The SNMP Management Framework presently consists of five major components:

- o An overall architecture, described in [RFC 2571](#) [8].
- 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](#) [9], STD 16, [RFC 1212](#) [10] and [RFC 1215](#) [11]. The second version, called SMIV2, is described in STD 58, [RFC 2578](#) [12], STD 58, [RFC 2579](#) [13] and STD 58, [RFC 2580](#) [14].

- 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](#) [15]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in [RFC 1901](#) [16] and [RFC 1906](#) [17]. The third version of the message protocol is called SNMPv3 and described in [RFC 1906](#) [17], [RFC 2572](#) [18] and [RFC 2574](#) [19].
- o Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in STD 15, [RFC 1157](#) [15]. A second set of protocol operations and associated PDU formats is described in [RFC 1905](#) [20].
- o A set of fundamental applications described in [RFC 2573](#) [21] and the view-based access control mechanism described in [RFC 2575](#) [22].

A more detailed introduction to the current SNMP Management Framework can be found in [RFC 2570](#) [23].

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 (use of Counter64). 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.

2. Revision History

This section will either be removed or considerably reduced when the draft is submitted for approval as a RFC.

Changes from [draft-ops-rfc2011-update-00.txt](#):

May 2002

Removed `ipv6InterfaceEffectiveMtu` and `ipv6InterfaceIdentifierLength`.

Added text to `ipaddressPrefixTable` to describe its utility.

Added text to `ipAddressTable` to state that multicast addresses are described in their own table(s).

Added `ipv4IfAdminStatus` and `ipv6InterfaceAdminStatus`.

Added text to `ipAddressPrefixOrigin` to describe that an address becomes well known by assignment from IANA or the address registries or by specification in a standards track RFC.

Added text to `ipaddressOrigin` to clarify the manual vs random difference.

Added text to `inetNetToMediaType` to clarify the difference between static and local.

Created textual conventions for the following `IpAddressOrigin`, `IpAddressStatus` and `IpAddressPrefixOrigin`.

Added persistence information to all read-write and read-create objects:

`ipForwarding` & `ipDefaultTTL` - should be persistent

`ipv6Forwarding`, `ipv4AdminStatus` & `ipv6InterfaceAdminStatus` - SHOULD be persistent

`ipv6InterfaceIdentifier` - modified from RW to Read-only

`ipNetToMediaPhysAddress` & `ipNetToMediaType` - should not be persistent

`inetNetToMediaPhysAddress` & `inetNetToMediaType` - SHOULD NOT be persistent

Added text to specify that `ifIndex` objects (`ipv4IfIndex`, `ipv6InterfaceIfIndex`, `ipIfStatsIfIndex`, `ipAddressPrefixIfIndex`, `ipAddressIfIndex` and `ipv6ScopeIdIfIndex`) use the same indexes as `ifIndex`.

Removed `ifindex` and code from the descriptions of `inetIcmpMsg{In Out}Pkts`.

Updated the text for the objects in the `ipIfStatsTable`. The new text allows them to be either system wide or interface specific. It also clarifies which interface should be used for some objects. Finally descriptions were added to some newer objects.

Added ipIfStatsRefreshRate

Modified the author information. The main author list has been moved to a section within the document and replaced on the front page with the editor's name. The MIB contact information has been modified.

Added text mentioning that ARP ([RFC826](#)) and ND ([RFC2461](#)) are the two most likely ways of populating the Net to Media table.

Added a reference clause to inetIcmpMsgType pointing to the IANA pages for ICMP and ICMPv6 parameters.

Added text mentioning that ipIfStatsInTooBigErrors is only valid for IPv6 but that it should be instantiated for IPv4 as well.

Added text to the ipAddressPrefixTable and it's objects mentioning that it isn't tuned for IPv4 and defaults to use for IPv4 addresses.

Added the ipAddressLastChanged object.

Added reference clauses to ipv6DefaultHopLimit and several objects in the ipAddressPrefix table. I don't think that any other objects that don't already have some text have something to reference.

Modified the IP statistics table.

ipIfStatsHCInUcastPkts became ipIfStatsHCInReceives

removed ipIfStatsInTooBigErrors and include those packets in ipIfStatsOutFragFails

added ipIfStatsInForwDatagrams to count packets that we attempt to forward

clarified the text in ipIfStatsOutForwDatagrams to make it clear that "success" was related to the forwarding step and not the transmission step

added ipIfStatsOutTransmits and ipIfStatsHCOutTransmits as counters of the packets sent to the lower layers and pointed the OutOctet counters to them

added ipIfStatsOutNoRoutes to count the number of locally generated datagrams that couldn't be transmitted as no route was found

added ipIfStatsOutFragReqds to count the number of datagrams that

require fragmentation

Added text and a Case diagram describing the statistics table.

Added the ipAddressCreated object.

Removed ipIfStatsHCOutUcastPkts as it HCOutTransmits replaces it.

Also re-arranged the statsTable sequence to try and group the objects more rationally.

Added reachable and retransmit times to the ipv6InterfaceTable

Added Default router list.

Added router advertisement configuration table. SARSARSAR

November 2001

Modified the ICMP message table to remove some indexes. The table no longer tracks counters with per-interface or per-ICMP code granularity.

12 Jul 2001

Changed to IPNG working group work item.

Removed mention of SIIT, since it's just for transition

Added lots of counters to ipIfStats table, and ipIfStatsDiscontinuityTime

Changed ipAddressIfIndex and ScopeIdentifier to refer to IF-MIB instead of [RFC 2863](#) or [RFC 2233](#)

Removed text about agents supporting a subset of values from ipv6Forwarding; this belongs in an AGENT-CAPABILITIES.

Un-deprecated ipReasmTimeout. XXX Do we need ipv6ReasmTimeout too? I think not; [RFC2460](#) seems to say that it's a constant 60 seconds.

Changes from first draft posted to v6mib mailing list:

23 Feb 2001

Added ipv4InterfaceTable

Added ipv6InterfaceTable

Added ipAddressPrefixTable and slightly reworked ipAddressTable (nee inetAddressTable).

Deprecated ipMIBCompliance. Still need to finish updated ones.

Added copyright and table of contents.

7 Feb 2001

Renamed inetIfStats to ipIfStats

Added ipv6ScopeTable

Added ScopeIdentifier TC, which should be in INET-ADDRESS-MIB.

Added SIZE to inetAddrAddr and inetNetToMediaAddress

Wrote some boilerplate for multi-interface-or-system-wide counter tables.

3. Overview

Add overview of the entire MIB, briefly describe each table and some of the interrelationships.

3.1. Multi-Stack Implementations

This MIB does not provide native support for implementations of multiple stacks sharing the same address family. One option for supporting such designs is to assign each stack within an address family to a separate context. These contexts could then be selected based upon the community string or context name, with the Entity MIB providing a method for listing the supported contexts.

3.2. Discussion of Tables and Groups

3.2.1. General Objects

IPv4

IPv6

3.2.2. IPv4 Interface Table

3.2.3. IPv6 Interface Table

3.2.4. IP Statistics Table The IP statistics table (ipIfStatsTable)

contains objects to count the number of datagrams and octets that a given entity has processed. Unlike the previous attempt this document uses a single table for multiple address families. Typically the only two families of interest are IPv4 and IPv6 however the table can support other families if necessary. There is also only one table to convey both system wide and interface specific information.

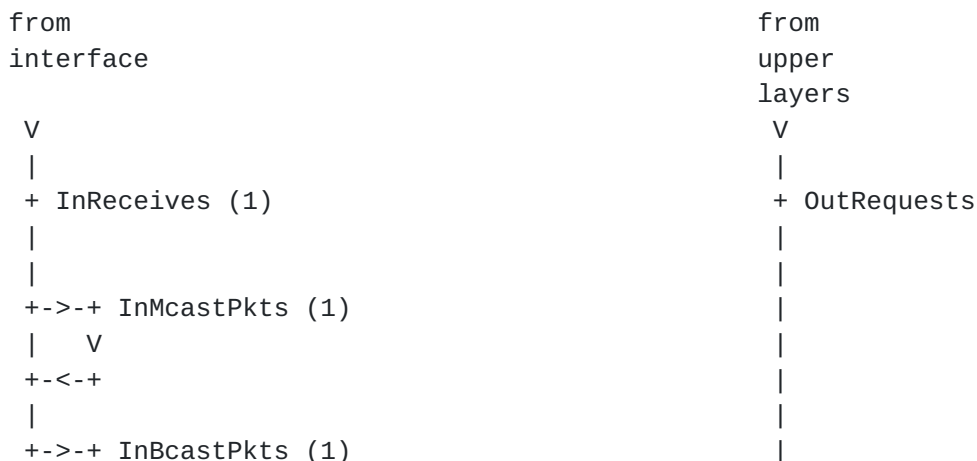
Both address family and interface information is encoded into the instance information used to index the table. The instance is formed from two sub-ids. The first represents the address family (IPv4 and IPv6) and the interface within that address family is represented by the second sub-id. If the interface instance equals the reserved value of zero (0) the entry represents system-wide information. That is the various counters are for all interfaces and not a specific set.

Two objects of note are ipIfStatsDiscontinuityTime and ipIfStatsRefreshRate. These objects provide information about the row in the table more than about the system itself.

The discontinuity object allows a management entity to determine if a discontinuity event which would invalidate the management entities understanding of the counters has occurred. The system being re-initialized or the interface being cycled are possible examples of a discontinuity event.

The refresh object allows a management entity to determine a proper polling interval for the rest of the objects.

The following Case diagram represents the general ordering of the packet counters. In order to avoid extra clutter the prefix "ipIfStats" has been removed from each of the counter names.





(1) The HC counters and octet counters are also found at these points but have been left out for clarity.

(2) The discard counters may increment at any time in the processing path.

(3) Local packets on the input side are counted on the interface associated with their destination address, which may not be the interface on which they were received. This requirement is caused by the possibility of losing the original interface during processing, especially re-assembly.

(4) Some re-assembly algorithms may lose track of the number of fragments during processing and so some fragments may not be counted in this object.

[3.2.5.](#) Internet Address Prefix Table

[3.2.6.](#) Internet Address Table

[3.2.7.](#) Internet Address Translation Table

[3.2.8.](#) IPv6 Scope Identifier

[3.2.9.](#) Default Router Table

[3.2.10.](#) ICMP Statistics Tables

General Counters

Message Specific Counters

[3.2.11.](#) Deprecated Objects

[3.2.12.](#) Conformance and Compliance

[4.](#) Updating Implementations

Boy, does this need more text. XXX

[4.1.](#) Updating an implementation of the IPv4-only IP-MIB

- o New counters: InTooBigErrors, InTruncatedPkts, InMcastPkts, OutMcastPkts.

- o Can report existing counters as system-wide in ipIfStatsTable (0 for ifindex) or can implement per-interface counters and report those.

- o Move ipAdEntReasmMaxSize to ipIfReasmMaxSize

4.2. Updating an implementation of the IPv6-MIB

- o New counters: ReasmTimeout
- o Don't need to implement per-interface stats.
- o Reimplement ipv6IfLowerLayer with the TUNNEL-MIB (note: can't represent IPv6-over-IPv6 since the TUNNEL-MIB is IPv4-only)

5. Definitions

IP-MIB DEFINITIONS ::= BEGIN

IMPORTS

```
MODULE-IDENTITY, OBJECT-TYPE,
Integer32, Counter32, IPAddress, mib-2, Unsigned32, Counter64
    FROM SNMPv2-SMI
PhysAddress, TruthValue, TimeStamp, RowPointer,
TEXTUAL-CONVENTION -- XXX
    FROM SNMPv2-TC
MODULE-COMPLIANCE, OBJECT-GROUP    FROM SNMPv2-CONF
InetAddress, InetAddressType,
InetAddressPrefixLength            FROM INET-ADDRESS-MIB
InterfaceIndex,
InterfaceIndexOrZero, ifIndex      FROM IF-MIB;
```

ipMIB MODULE-IDENTITY

```
LAST-UPDATED "200107130000Z"
ORGANIZATION "IETF IPv6 MIB Revision Team"
CONTACT-INFO
    "Editor:
     Shawn A. Routhier
     Wind River
     500 Wind River Way
     Alameda, CA
     USA

     Phone: +1 510 749 2095
     EMail: <sar@epilogue.com>"
```

DESCRIPTION

"The MIB module for managing IP and ICMP implementations, but excluding their management of IP routes."

REVISION "200207010000Z"

DESCRIPTION

"Update and clarify the IP version neutral revision adding


```
        the ND objects."
REVISION      "200107130000Z"
DESCRIPTION
    "IP version neutral revision, published as RFC XXXX."
REVISION      "9411010000Z"
DESCRIPTION
    "Published seperately as RFC 2011."
REVISION      "9103310000Z"
DESCRIPTION
    "The initial revision of this MIB module was part of MIB-II."
 ::= { mib-2 48}

--
-- The textual conventions we define and use in this MIB.
--

IpAddressOrigin ::= TEXTUAL-CONVENTION
    STATUS      current
    DESCRIPTION
        "The origin of the address.

        manual(2) indicates that the address was manually configured
        to a specified address, e.g by user configuration.

        wellknown(3) indicates an address constructed from a well-
        known value, e.g. an IANA-assigned anycast address.

        dhcp(4) indicates an address that was assigned to this
        system by a DHCP server.

        linklayer(5) indicates an address created by IPv6 stateless
        autoconfiguration.

        random(6) indicates an address chosen by the system at
        random, e.g. an IPv4 address within 169.254/16, or an RFC
        3041 privacy address."
    SYNTAX      INTEGER {
        other(1),
        manual(2),
        wellknown(3),
        dhcp(4),      -- XXX or assignedbyserver ?
        linklayer(5),
        random(6)
    }

IpAddressStatus ::= TEXTUAL-CONVENTION
    STATUS      current
    DESCRIPTION
```


"The status of an address. Most of the states correspond to states from the IPv6 Stateless Address Autoconfiguration protocol [5]

The preferred(1) state indicates that this is a valid address that can appear as the destination or source address of a packet.

The deprecated(2) state indicates that this is a valid but deprecated address that should no longer be used as a source address in new communications, but packets addressed to such an address are processed as expected.

The invalid(3) state indicates that this is not valid address which should not appear as the destination or source address of a packet.

The inaccessible(4) state indicates that the address is not accessible because the interface to which this address is assigned is not operational.

The unknown(5) state indicates that the status can not be determined for some reason.

The tentative(6) state indicates the uniqueness of the address on the link is being verified. Addresses in this state should not be used for general communication and should only be used to determine the uniqueness of the address.

The duplicate(7) state indicates the address has been determined to be non-unique on the link and so must not be used.

In the absence of other information, an IPv4 address is always preferred(1)."

```
SYNTAX      INTEGER {  
    preferred(1),  
    deprecated(2),  
    invalid(3),  
    inaccessible(4),  
    unknown(5),  
    tentative(6),  
    duplicate(7)  
}
```

IpAddressPrefixOrigin ::= TEXTUAL-CONVENTION

DESCRIPTION

"The origin of this prefix.

manual(2) indicates a prefix that was manually configured.

wellknown(3) indicates a well-known prefix, e.g. 169.254/16 for IPv4 autoconfiguration or fe80::/10 for IPv6 link-local addresses. Well known prefixes may be assigned by IANA or the address registries or by specification in a standards track RFC.

dhcp(4) indicates a prefix that was assigned by a DHCP server.

routeradv(5) indicates a prefix learned from a router advertisement.

Note: while IPAddressOrigin and IPAddressPrefixOrigin are similar they are not identical. The first defines how an address was created while the second defines how a prefix was found."

```
SYNTAX      INTEGER {
    other(1),
    manual(2),
    wellknown(3),
    dhcp(4),
    routeradv(5)
}
```

```
--
-- the IP general group
-- some objects that affect all of IPv4
--
```

```
ip          OBJECT IDENTIFIER ::= { mib-2 4 }
```

```
ipForwarding OBJECT-TYPE
```

```
SYNTAX      INTEGER {
                                forwarding(1),    -- acting as a router
                                notForwarding(2)  -- NOT acting as a router
                            }

```

```
MAX-ACCESS  read-write
```

```
STATUS      current
```

```
DESCRIPTION
```

"The indication of whether this entity is acting as an IPv4 router in respect to the forwarding of datagrams received by, but not addressed to, this entity. IPv4 routers forward datagrams. IPv4 hosts do not (except those source-routed

via the host).

When this object is written the entity should save the change to non-volatile storage and restore the object from non-volatile storage upon re-initilaization of the system.
Note: a stronger requirement is not used because this object was previously defined."

::= { ip 1 }

ipDefaultTTL OBJECT-TYPE

SYNTAX INTEGER (1..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The default value inserted into the Time-To-Live field of the IPv4 header of datagrams originated at this entity, whenever a TTL value is not supplied by the transport layer protocol.

When this object is written the entity should save the change to non-volatile storage and restore the object from non-volatile storage upon re-initilaization of the system.
Note: a stronger requirement is not used because this object was previously defined."

::= { ip 2 }

ipReasmTimeout OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The maximum number of seconds which received fragments are held while they are awaiting reassembly at this entity."

::= { ip 13 }

--

-- the IPv6 general group

-- Some objects that affect all of IPv6

--

ipv6MIB OBJECT IDENTIFIER ::= { mib-2 55 }

ipv6MIBObjects OBJECT IDENTIFIER ::= { ipv6MIB 1 }

ipv6Forwarding OBJECT-TYPE

SYNTAX INTEGER {

forwarding(1), -- acting as a router

notForwarding(2) -- NOT acting as a router


```
    }
MAX-ACCESS read-write
STATUS      current
DESCRIPTION
    "The indication of whether this entity is acting as an IPv6
    router in respect to the forwarding of datagrams received
    by, but not addressed to, this entity.  IPv6 routers forward
    datagrams.  IPv6 hosts do not (except those source-routed
    via the host).

    When this object is written the entity SHOULD save the
    change to non-volatile storage and restore the object from
    non-volatile storage upon re-initilaization of the system."
 ::= { ipv6MIBObjects 1 }
```

```
ipv6DefaultHopLimit OBJECT-TYPE
SYNTAX      INTEGER (0..255)
MAX-ACCESS read-write
STATUS      current
DESCRIPTION
    "The default value inserted into the Hop Limit field of the
    IPv6 header of datagrams originated at this entity, whenever
    a Hop Limit value is not supplied by the transport layer
    protocol.

    When this object is written the entity SHOULD save the
    change to non-volatile storage and restore the object from
    non-volatile storage upon re-initilaization of the system."
REFERENCE "RFC2461 Section 6.3.2"
 ::= { ipv6MIBObjects 2 }
```

```
--
-- IPv4 Interface Table
--
```

```
ipv4IfTable OBJECT-TYPE
SYNTAX      SEQUENCE OF Ipv4IfEntry
MAX-ACCESS not-accessible
STATUS      current
DESCRIPTION
    "The table containing per-interface IPv4-specific
    information."
 ::= { ip 25 }
```

```
ipv4IfEntry OBJECT-TYPE
SYNTAX      Ipv4IfEntry
MAX-ACCESS not-accessible
STATUS      current
```


DESCRIPTION

"An entry containing IPv4-specific information for a specific interface."

INDEX { ipv4IfIndex }

::= { ipv4IfTable 1 }

Ipv4IfEntry ::= SEQUENCE {
 ipv4IfIndex InterfaceIndex,
 ipv4IfReasmMaxSize Integer32,
 ipv4IfAdminStatus INTEGER
}

ipv4IfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The index value which uniquely identifies the interface to which this entry is applicable. The interface identified by a particular value of this index is the same interface as identified by the same value of the IF-MIB's ifIndex."

::= { ipv4IfEntry 1 }

ipv4IfReasmMaxSize OBJECT-TYPE

SYNTAX Integer32 (0..65535)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The size of the largest IPv4 datagram which this entity can re-assemble from incoming IPv4 fragmented datagrams received on this interface."

::= { ipv4IfEntry 2 }

ipv4IfAdminStatus OBJECT-TYPE

SYNTAX INTEGER {
 up(1),
 down(2)
}

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The indication of whether IPv4 is enabled (up) or disabled (down) on this interface. This object does not affect the state of the interface itself, only its connection to an IPv4 stack. The IF-MIB should be used to control the state of the interface.

When this object is written the entity SHOULD save the


```
        change to non-volatile storage and restore the object from
        non-volatile storage upon re-initilaization of the system."
 ::= { ipv4IfEntry 3 }
```

```
--
```

```
-- v6 interface table
```

```
-- XXX I suspect that most of these objects can go away.
```

```
--
```

```
ipv6InterfaceTable OBJECT-TYPE
```

```
    SYNTAX      SEQUENCE OF Ipv6InterfaceEntry
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "The table containing per-interface IPv6-specific
        information."
```

```
 ::= { ip 31 }
```

```
ipv6InterfaceEntry OBJECT-TYPE
```

```
    SYNTAX      Ipv6InterfaceEntry
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "An entry containing IPv6-specific information for a given
        interface."
```

```
    INDEX { ipv6InterfaceIfIndex }
```

```
 ::= { ipv6InterfaceTable 1 }
```

```
Ipv6InterfaceEntry ::= SEQUENCE {
```

```
    ipv6InterfaceIfIndex      InterfaceIndex,
```

```
    ipv6InterfaceReasmMaxSize  Unsigned32,
```

```
    ipv6InterfaceIdentifier    Ipv6AddressIfIdentifier,
```

```
    ipv6InterfacePhysicalAddress PhysAddress,
```

```
    ipv6InterfaceAdminStatus   INTEGER,
```

```
    ipv6IntefaceReachableTime  Unsigned32,
```

```
    ipv6IntefaceRetransmitTime  Unsigned32
```

```
}
```

```
ipv6InterfaceIfIndex OBJECT-TYPE
```

```
    SYNTAX      InterfaceIndex
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "The index value which uniquely identifies the interface to
        which this entry is applicable.  The interface identified by
        a particular value of this index is the same interface as
```



```
        identified by the same value of the IF-MIB's ifIndex."
 ::= { ipv6InterfaceEntry 1 }

-- ipv6InterfaceEffectiveMtu
-- was      ::= { ipv6InterfaceEntry 2 }
-- XXX renumber?

ipv6InterfaceReasmMaxSize OBJECT-TYPE
    SYNTAX      Unsigned32 (0..65535)
    UNITS        "octets"
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The size of the largest IPv6 datagram which this entity can
         re-assemble from incoming IPv6 fragmented datagrams received
         on this interface."
    ::= { ipv6InterfaceEntry 3 }

-- XXX ugh: I want to get rid of this, which is why it's in the middle
-- of nowhere
Ipv6AddressIfIdentifier ::= TEXTUAL-CONVENTION
    DISPLAY-HINT "2x:"
    STATUS       current
    DESCRIPTION
        "This data type is used to model IPv6 address
         interface identifiers. This is a binary string
         of up to 8 octets in network byte-order."
    SYNTAX       OCTET STRING (SIZE (0..8))

ipv6InterfaceIdentifier OBJECT-TYPE
    SYNTAX       Ipv6AddressIfIdentifier
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The Interface Identifier for this interface that is (at
         least) unique on the link this interface is attached to. The
         Interface Identifier is combined with an address prefix to
         form an interface address.

         By default, the Interface Identifier is autoconfigured
         according to the rules of the link type this interface is
         attached to.

         XXX - is this an EUI64 that belongs more in the IF-MIB?"
    ::= { ipv6InterfaceEntry 4 }

-- ipv6InterfaceIdentifierLength
-- was      ::= { ipv6InterfaceEntry 5 }
```


-- XXX renumber?

ipv6InterfacePhysicalAddress OBJECT-TYPE

SYNTAX PhysAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The interface's physical address. For example, for an IPv6 interface attached to an 802.x link, this object normally contains a MAC address. Note that in some cases this address may differ from the address of the interface's protocol sub-layer. The interface's media-specific MIB must define the bit and byte ordering and the format of the value of this object. For interfaces which do not have such an address (e.g., a serial line), this object should contain an octet string of zero length.

XXX When can this be different from the address of the interface's protocol sub-layer, and why?"

::= { ipv6InterfaceEntry 6 }

ipv6InterfaceAdminStatus OBJECT-TYPE

SYNTAX INTEGER {
 up(1),
 down(2)
}

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The indication of whether IPv6 is enabled (up) or disabled (down) on this interface. This object does not affect the state of the interface itself, only its connection to an IPv4 stack. The IF-MIB should be used to control the state of the interface.

When this object is written the entity SHOULD save the change to non-volatile storage and restore the object from non-volatile storage upon re-initilaization of the system."

::= { ipv6InterfaceEntry 7 }

ipv6InterfaceReachableTime OBJECT-TYPE

SYNTAX Unsigned32

UNITS "milliseconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The time a neighbor is considered reachable after receiving a reachability confirmation."

REFERENCE "[RFC2461, Section 6.3.2](#)"

::= { ipv6InterfaceEntry 8 }

ipv6InterfaceRetransmitTime OBJECT-TYPE

SYNTAX Unsigned32

UNITS "milliseconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The time between retransmissions of Neighbor Solicitation messages to a neighbor when resolving the address or when probing the reachability of a neighbor."

REFERENCE "[RFC2461, Section 6.3.2](#)"

--

-- Per-Interface or System-Wide IP statistics.

--

ipIfStatsTable OBJECT-TYPE

SYNTAX SEQUENCE OF IpIfStatsEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The table containing traffic statistics. These statistics may be kept per-interface and/or system-wide."

::= { ip 26 }

ipIfStatsEntry OBJECT-TYPE

SYNTAX IpIfStatsEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An interface statistics entry containing objects for a particular interface, or system-wide.

A row with an ipIfStatsIfIndex value of zero indicates a system-wide value; a row with a non-zero ipIfStatsIfIndex indicates an interface-specific value. A system may provide both system-wide and interface-specific values, in which case it is important to note that the system-wide value may not be equal to the sum of the interface-specific values across all interfaces due to e.g. dynamic interface creation/deletion."

INDEX { ipIfStatsAFType, ipIfStatsIfIndex }

::= { ipIfStatsTable 1 }


```
IpIfStatsEntry ::= SEQUENCE {  
    ipIfStatsAftype          InetAddressType,  
    ipIfStatsIfIndex         InterfaceIndexOrZero,  
    ipIfStatsInReceives      Counter32,  
    ipIfStatsHCInReceives    Counter64,  
    ipIfStatsInOctets         Counter32,  
    ipIfStatsHCInOctets      Counter64,  
    ipIfStatsInHdrErrors     Counter32,  
    ipIfStatsInNoRoutes      Counter32,  
    ipIfStatsInAddrErrors    Counter32,  
    ipIfStatsInUnknownProtos Counter32,  
    ipIfStatsInTruncatedPkts Counter32,  
    ipIfStatsInForwDatagrams Counter32,  
    ipIfStatsReasmReqds       Counter32,  
    ipIfStatsReasmOKs         Counter32,  
    ipIfStatsReasmFails       Counter32,  
    ipIfStatsInDiscards       Counter32,  
    ipIfStatsInDelivers       Counter32,  
    ipIfStatsOutRequests     Counter32,  
    ipIfStatsOutNoRoutes     Counter32,  
    ipIfStatsOutForwDatagrams Counter32,  
    ipIfStatsOutDiscards     Counter32,  
    ipIfStatsOutFragReqds     Counter32,  
    ipIfStatsOutFragOKs       Counter32,  
    ipIfStatsOutFragFails     Counter32,  
    ipIfStatsOutFragCreates   Counter32,  
    ipIfStatsOutTransmits     Counter32,  
    ipIfStatsHCOutTransmits   Counter64,  
    ipIfStatsOutOctets        Counter32,  
    ipIfStatsHCOutOctets      Counter64,  
    ipIfStatsInMcastPkts     Counter32,  
    ipIfStatsHCInMcastPkts   Counter64,  
    ipIfStatsInMcastOctets    Counter32,  
    ipIfStatsHCInMcastOctets Counter64,  
    ipIfStatsOutMcastPkts     Counter32,  
    ipIfStatsHCOutMcastPkts   Counter64,  
    ipIfStatsOutMcastOctets   Counter32,  
    ipIfStatsHCOutMcastOctets Counter64,  
    ipIfStatsInBcastPkts     Counter32,  
    ipIfStatsHCInBcastPkts   Counter64,  
    ipIfStatsOutBcastPkts     Counter32,  
    ipIfStatsHCOutBcastPkts   Counter64,  
    ipIfStatsDiscontinuityTime Timestamp,  
    ipIfStatsRefreshRate      Unsigned32  
}
```

```
ipIfStatsAftype OBJECT-TYPE  
    SYNTAX      InetAddressType
```


MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The address family for this row. May only be IPv4 or IPv6."

::= { ipIfStatsEntry 1 }

ipIfStatsIfIndex OBJECT-TYPE

SYNTAX InterfaceIndexOrZero

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The index value which uniquely identifies the interface to which this entry is applicable. The interface identified by a particular value of this index is the same interface as identified by the same value of the IF-MIB's ifIndex.

A value of zero is used for system-wide counters."

::= { ipIfStatsEntry 2 }

ipIfStatsInReceives OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of input IP datagrams received, including those received in error."

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 3 }

ipIfStatsHCInReceives OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of input IP datagrams received, including those received in error. This object counts the same datagrams as ipIfStatsInReceives but allows for larger values.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 4 }

ipIfStatsInOctets OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of octets received in input IP datagrams, including those received in error. Octets from datagrams counted in ipIfStatsInReceives MUST be counted here.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 5 }

ipIfStatsHCInOctets OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of octets received in input IP datagrams, including those received in error. This object counts the same octets as ipIfStatsInOctets but allows for larger values.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 6 }

ipIfStatsInHdrErrors OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of input IP datagrams discarded due to errors in their IP headers, including version number mismatch, other format errors, hop count exceeded, errors discovered in processing their IP options, etc.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 7 }

-- ipIfStatsInTooBigErrors


```
-- was      ::= { ipIfStatsEntry 8 }  
-- XXX renumber?
```

ipIfStatsInNoRoutes OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of input IP datagrams discarded because no route could be found to transmit them to their destination.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 9 }

ipIfStatsInAddrErrors OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of input IP datagrams discarded because the IP address in their IP header's destination field was not a valid address to be received at this entity. This count includes invalid addresses (e.g., ::0) and unsupported addresses (e.g., addresses with unallocated prefixes). For entities which are not IP routers and therefore do not forward datagrams, this counter includes datagrams discarded because the destination address was not a local address.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 10 }

ipIfStatsInUnknownProtos OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of locally-addressed IP datagrams received successfully but discarded because of an unknown or unsupported protocol.

When tracking interface statistics the counter of the interface to which these datagrams were addressed is

incremented. This interface might not be the same as the input interface for some of the datagrams.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of
ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 11 }

ipIfStatsInTruncatedPkts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of input IP datagrams discarded because datagram frame didn't carry enough data.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of
ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 12 }

ipIfStatsInForwDatagrams OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of input datagrams for which this entity was not their final IP destination and for which this entity attempted to find a route to forward them to that final destination. In entities which do not act as IP routers, this counter will include only those datagrams which were Source-Routed via this entity, and the Source-Route processing was successful.

When tracking interface statistics the counter of the incoming interface is incremented for each datagram.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of
ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 13 }

ipIfStatsReasmReqds OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP fragments received which needed to be reassembled at this interface.

When tracking interface statistics the counter of the interface to which these fragments were addressed is incremented. This interface might not be the same as the input interface for some of the fragments.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 14 }

ipIfStatsReasmOKs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP datagrams successfully reassembled.

When tracking interface statistics the counter of the interface to which these datagrams were addressed is incremented. This interface might not be the same as the input interface for some of the datagrams.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 15 }

ipIfStatsReasmFails OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of failures detected by the IP re-assembly algorithm (for whatever reason: timed out, errors, etc.). Note that this is not necessarily a count of discarded IP fragments since some algorithms (notably the algorithm in [RFC 815](#)) can lose track of the number of fragments by combining them as they are received.

When tracking interface statistics the counter of the interface to which these fragments were addressed is

incremented. This interface might not be the same as the input interface for some of the fragments.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 16 }

ipIfStatsInDiscards OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of input IP datagrams for which no problems were encountered to prevent their continued processing, but which were discarded (e.g., for lack of buffer space). Note that this counter does not include any datagrams discarded while awaiting re-assembly.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 17 }

ipIfStatsInDelivers OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of datagrams successfully delivered to IP user-protocols (including ICMP).

When tracking interface statistics the counter of the interface to which these datagrams were addressed is incremented. This interface might not be the same as the input interface for some of the datagrams.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 18 }

ipIfStatsOutRequests OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of IP datagrams which local IP user-protocols (including ICMP) supplied to IP in requests for transmission. Note that this counter does not include any datagrams counted in ipIfStatsOutForwDatagrams.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 19 }

ipIfStatsOutNoRoutes OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of local generated IP datagrams discarded because no route could be found to transmit them to their destination.

As no route can be found for these datagrams the interface specific instances are not meaningful for this object.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 20 }

ipIfStatsOutForwDatagrams OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of datagrams which this entity received and for which it was successful in finding a path to their final destination. In entities which do not act as IP routers, this counter will include only those datagrams which were Source-Routed via this entity, and the Source-Route processing was successful.

When tracking interface statistics the counter of the outgoing interface is incremented for a successfully forwarded datagram.

Discontinuities in the value of this counter can occur at

re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 21 }

ipIfStatsOutDiscards OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of output IP datagrams for which no problem was encountered to prevent their transmission to their destination, but which were discarded (e.g., for lack of buffer space). Note that this counter would include datagrams counted in ipIfStatsOutForwDatagrams if any such datagrams met this (discretionary) discard criterion.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 22 }

ipIfStatsOutFragReqds OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP datagrams that would require fragmentation in order to be transmitted.

When tracking interface statistics the counter of the outgoing interface is incremented for a successfully fragmented datagram.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 23 }

ipIfStatsOutFragOKs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP datagrams that have been successfully fragmented.

When tracking interface statistics the counter of the outgoing interface is incremented for a successfully fragmented datagram.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of
ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 24 }

ipIfStatsOutFragFails OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP datagrams that have been discarded because they needed to be fragmented but could not be. This includes IPv4 packets that have the DF bit set and IPv6 packets that are being forwarded and exceed the outgoing link MTU.

When tracking interface statistics the counter of the outgoing interface is incremented for an unsuccessfully fragmented datagram.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of
ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 25 }

ipIfStatsOutFragCreates OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of output datagram fragments that have been generated as a result of IP fragmentation.

When tracking interface statistics the counter of the outgoing interface is incremented for a successfully fragmented datagram.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of
ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 26 }

ipIfStatsOutTransmits OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of IP datagrams that this entity supplied to the lower layers for transmission. This includes datagrams generated local and those forwarded by this entity.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 27 }

ipIfStatsHCOutTransmits OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of IP datagrams that this entity supplied to the lower layers for transmission. This object counts the same datagrams as ipIfStatsOutTransmits but allows for larger values.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 28 }

ipIfStatsOutOctets OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of octets in IP datagrams delivered to the lower layers for transmission. Octets from datagrams counted in ipIfStatsOutTransmits MUST be counted here.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 29 }

ipIfStatsHCOutOctets OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of octets in IP datagrams delivered to the lower layers for transmission. This object counts the same octets as ipIfStatsOutOctets but allows for larger values.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 30 }

ipIfStatsInMcastPkts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP multicast datagrams received.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 31 }

ipIfStatsHCInMcastPkts OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP multicast datagrams received. This object counts the same datagrams as ipIfStatsInMcastPkts but allows for larger values.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 32 }

ipIfStatsInMcastOctets OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of octets received in IP multicast

datagrams. Octets from datagrams counted in ipIfStatsOutMcastPkts MUST be counted here.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 33 }

ipIfStatsHCInMcastOctets OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of octets received in IP multicast datagrams. This object counts the same octets as ipIfStatsInMcastOctets but allows for larger values.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 34 }

ipIfStatsOutMcastPkts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP multicast datagramss transmitted.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 35 }

ipIfStatsHCOutMcastPkts OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP multicast datagrams transmitted. This object counts the same datagrams as ipIfStatsOutMcastPkts but allows for larger values.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other

times as indicated by the value of
ipIfStatsDiscontinuityTime."
::= { ipIfStatsEntry 36 }

ipIfStatsOutMcastOctets OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of octets transmitted in IP multicast datagrams. Octets from datagramss counted in ipIfStatsInMcastPkts MUST be counted here.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 37 }

ipIfStatsHCOutMcastOctets OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of octets received in IP multicast datagrams. This object counts the same octets as ipIfStatsOutMcastOctets but allows for larger values.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 38 }

ipIfStatsInBcastPkts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP broadcast datagrams received.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 39 }

ipIfStatsHCInBcastPkts OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP broadcast datagrams received. This object counts the same datagrams as ipIfStatsInBcastPkts but allows for larger values.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 40 }

ipIfStatsOutBcastPkts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP broadcast datagrams transmitted.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 41 }

ipIfStatsHCOutBcastPkts OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP broadcast datagrams transmitted. This object counts the same datagrams as ipIfStatsOutBcastPkts but allows for larger values.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ipIfStatsDiscontinuityTime."

::= { ipIfStatsEntry 42 }

ipIfStatsDiscontinuityTime OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime on the most recent occasion at which

any one or more of this entry's counters suffered a discontinuity.

If no such discontinuities have occurred since the last reinitialization of the local management subsystem, then this object contains a zero value."

::= { ipIfStatsEntry 43 }

ipIfStatsRefreshRate OBJECT-TYPE

SYNTAX Unsigned32

UNITS "milli-seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The minimum reasonable polling interval for this entry.

This object provides an indication of the minimum amount of time required to update the counters in this entry."

::= { ipIfStatsEntry 44 }

--

-- Internet Address Prefix table

--

-- Open Issues:

-- Is there a better SMI data type for *Lifetime objects?

ipAddressPrefixTable OBJECT-TYPE

SYNTAX SEQUENCE OF IpAddressPrefixEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The inet prefix table.

This table allows the user to determine the source of an IP address or set of IP addresses and allows other tables to share the information via pointer rather than by copying.

For example when the node configures both a unicast and anycast address for a prefix the ipAddressPrefix objects for those addresses will point to a single row in this table.

This table is primarily provides support for IPv6 prefixes and several of the objects are less meaningful for IPv4. The table continues to allow IPv4 addresses to allow future flexibility. In order to promote a common configuration this document includes suggestions for default values for IPv4 prefixes. Each of these values may be overridden if an

object is meaningful to the node."
::= { ip 27 }

ipAddressPrefixEntry OBJECT-TYPE

SYNTAX IpAddressPrefixEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"inet prefix entry"

INDEX { ipAddressPrefixIfIndex, ipAddressPrefixType,
 ipAddressPrefixPrefix, ipAddressPrefixLength }

::= { ipAddressPrefixTable 1 }

IpAddressPrefixEntry ::= SEQUENCE {

ipAddressPrefixIfIndex	InterfaceIndex,
ipAddressPrefixType	InetAddressType,
ipAddressPrefixPrefix	InetAddress,
ipAddressPrefixLength	InetAddressPrefixLength,
ipAddressPrefixOrigin	IpAddressPrefixOrigin,
ipAddressPrefixOnLinkFlag	TruthValue,
ipAddressPrefixAutonomousFlag	TruthValue,
ipAddressPrefixAdvPreferredLifetime	Unsigned32,
ipAddressPrefixAdvValidLifetime	Unsigned32

}

ipAddressPrefixIfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The index value which uniquely identifies the interface on which this prefix is configured. The interface identified by a particular value of this index is the same interface as identified by the same value of the IF-MIB's ifIndex."

::= { ipAddressPrefixEntry 1 }

ipAddressPrefixType OBJECT-TYPE

SYNTAX InetAddressType

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The address type of ipAddressPrefix. Only IPv4 and IPv6 addresses are expected."

::= { ipAddressPrefixEntry 2 }

ipAddressPrefixPrefix OBJECT-TYPE

SYNTAX InetAddress (SIZE(0..36))

MAX-ACCESS not-accessible

STATUS current
DESCRIPTION
 "The address prefix. Bits after ipAddressPrefixLength must
 be zero."
::= { ipAddressPrefixEntry 3 }

ipAddressPrefixLength OBJECT-TYPE
SYNTAX InetAddressPrefixLength
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "The prefix length associated with this prefix."
::= { ipAddressPrefixEntry 4 }

ipAddressPrefixOrigin OBJECT-TYPE
SYNTAX IpAddressPrefixOrigin
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "The origin of this prefix."
::= { ipAddressPrefixEntry 5 }

ipAddressPrefixOnLinkFlag OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "This object has the value 'true(1)', if this prefix can be
 used for on-link determination and the value 'false(2)'
 otherwise.

 The default for IPv4 prefixes is 'true(1)'."
REFERENCE "For IPv6 [RFC2461](#), especially sections [2](#) and [4.6.2](#) and [RFC2462](#)"
::= { ipAddressPrefixEntry 6 }

ipAddressPrefixAutonomousFlag OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "Autonomous address configuration flag. When true(1),
 indicates that this prefix can be used for autonomous
 address configuration (i.e. can be used to form a local
 interface address). If false(2), it is not used to
 autoconfigure a local interface address.

 The default for IPv4 prefixes is 'false(2)'."
REFERENCE "For IPv6 [RFC2461](#), especially sections [2](#) and [4.6.2](#) and [RFC2462](#)"


```
::= { ipAddressPrefixEntry 7 }
```

ipAddressPrefixAdvPreferredLifetime OBJECT-TYPE

SYNTAX Unsigned32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The length of time in seconds that this prefix will remain preferred, i.e. time until deprecation. A value of 4,294,967,295 represents infinity.

The address generated from a deprecated prefix should no longer be used as a source address in new communications, but packets received on such an interface are processed as expected.

The default for IPv4 prefixes is 4,294,967,295 (infinity)."

REFERENCE "For IPv6 [RFC2461](#), especially sections [2](#) and [4.6.2](#) and [RFC2462](#)"

```
::= { ipAddressPrefixEntry 8 }
```

ipAddressPrefixAdvValidLifetime OBJECT-TYPE

SYNTAX Unsigned32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The length of time, in seconds, that this prefix will remain valid, i.e. time until invalidation. A value of 4,294,967,295 represents infinity.

The address generated from an invalidated prefix should not appear as the destination or source address of a packet.

The default for IPv4 prefixes is 4,294,967,295 (infinity)."

REFERENCE "For IPv6 [RFC2461](#), especially sections [2](#) and [4.6.2](#) and [RFC2462](#)"

```
::= { ipAddressPrefixEntry 9 }
```

--

-- Internet Address Table

--

-- Open Issues:

-- should ipAddressV4BcastAddr go somewhere else?

-- meeting notes said: dave: pointer to prefix table. What's that mean?

ipAddressTable OBJECT-TYPE

SYNTAX SEQUENCE OF IpAddressEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"inet addr table

This table contains addressing information relevant to the entity's interfaces.

This table does not contain multicast address information. Tables for such information should be contained in multicast specific MIBs such as [RFC3019](#)."

::= { ip 28 }

ipAddressEntry OBJECT-TYPE

SYNTAX IpAddressEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"inet addr entry"

INDEX { ipAddressAddrType, ipAddressAddr }

::= { ipAddressTable 1 }

IpAddressEntry ::= SEQUENCE {

ipAddressAddrType InetAddressType,

ipAddressAddr InetAddress,

ipAddressIfIndex InterfaceIndex,

ipAddressType INTEGER,

ipAddressPrefix RowPointer,

ipAddressOrigin IpAddressOrigin,

ipAddressStatus IpAddressStatus,

ipAddressCreated TimeStamp,

ipAddressLastChanged TimeStamp

}**ipAddressAddrType OBJECT-TYPE**

SYNTAX InetAddressType

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The address type of ipAddressAddr."

::= { ipAddressEntry 1 }

ipAddressAddr OBJECT-TYPE

SYNTAX InetAddress (SIZE(0..36))

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The IP address to which this entry's addressing information pertains."

::= { ipAddressEntry 2 }

ipAddressIfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The index value which uniquely identifies the interface to which this entry is applicable. The interface identified by a particular value of this index is the same interface as identified by the same value of the IF-MIB's ifIndex."

::= { ipAddressEntry 3 }

ipAddressType OBJECT-TYPE

SYNTAX INTEGER {
 unicast(1),
 anycast(2),
 broadcast(3)

}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The type of address. broadcast(3) is not a valid value for IPv6 addresses [[draft-ietf-ipngwg-addr-arch-v3-05.txt](#)]. "

::= { ipAddressEntry 4 }

ipAddressPrefix OBJECT-TYPE

SYNTAX RowPointer

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A pointer to the row in the prefix table to which this address belongs. May be { 0 0 } if there is no such row."

::= { ipAddressEntry 5 }

ipAddressOrigin OBJECT-TYPE

SYNTAX IpAddressOrigin

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The origin of the address."

::= { ipAddressEntry 6 }

ipAddressStatus OBJECT-TYPE

SYNTAX IpAddressStatus

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The status of the address, describing if the address can be used for communication.

In the absence of other information, an IPv4 address is always preferred(1)."

::= { ipAddressEntry 7 }

ipAddressCreated OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime at the time this entry was created. If this entry was created prior to the last re-initialization of the local network management subsystem, then this object contains a zero value."

::= { ipAddressEntry 8 }

ipAddressLastChanged OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime at the time this entry was last updated. If this entry was updated prior to the last re-initialization of the local network management subsystem, then this object contains a zero value."

::= { ipAddressEntry 9 }

--
-- the Internet Address Translation table
--

inetNetToMediaTable OBJECT-TYPE

SYNTAX SEQUENCE OF InetNetToMediaEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The IP Address Translation table used for mapping from IP addresses to physical addresses.

The Address Translation tables contain the IP address to 'physical' address equivalences. Some interfaces do not use

translation tables for determining address equivalences (e.g., DDN-X.25 has an algorithmic method); if all interfaces are of this type, then the Address Translation table is empty, i.e., has zero entries.

While many protocols may be used to populate this table, ARP [1] and Neighbor Discovery [4] are the most likely options."

::= { ip 29 }

inetNetToMediaEntry OBJECT-TYPE

SYNTAX InetNetToMediaEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Each entry contains one IP address to `physical' address equivalence."

INDEX { ifIndex,
inetNetToMediaNetAddressType,
inetNetToMediaNetAddress }

::= { inetNetToMediaTable 1 }

InetNetToMediaEntry ::= SEQUENCE {

inetNetToMediaNetAddressType InetAddressType,
inetNetToMediaNetAddress InetAddress,
inetNetToMediaPhysAddress PhysAddress,
inetNetToMediaLastUpdated TimeStamp,
inetNetToMediaType INTEGER,
inetNetToMediaState INTEGER

}

inetNetToMediaNetAddressType OBJECT-TYPE

SYNTAX InetAddressType

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The type of inetNetToMediaNetAddress."

::= { inetNetToMediaEntry 1 }

inetNetToMediaNetAddress OBJECT-TYPE

SYNTAX InetAddress (SIZE(0..36))

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The IP Address corresponding to the media-dependent
`physical' address."

::= { inetNetToMediaEntry 2 }

inetNetToMediaPhysAddress OBJECT-TYPE

SYNTAX PhysAddress

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The media-dependent `physical' address.

As the entries in this table are typically not persistent when this object is written the entity SHOULD NOT save the change to non-volatile storage."

::= { inetNetToMediaEntry 3 }

inetNetToMediaLastUpdated OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime at the time this entry was last updated. If this entry was updated prior to the last re-initialization of the local network management subsystem, then this object contains a zero value."

::= { inetNetToMediaEntry 4 }

inetNetToMediaType OBJECT-TYPE

SYNTAX INTEGER {
 other(1), -- none of the following
 invalid(2), -- an invalidated mapping
 dynamic(3),
 static(4),
 local(5) -- local interface
}

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The type of mapping.

Setting this object to the value invalid(2) has the effect of invalidating the corresponding entry in the inetNetToMediaTable. That is, it effectively disassociates the interface identified with said entry from the mapping identified with said entry. It is an implementation-specific matter as to whether the agent removes an invalidated entry from the table. Accordingly, management stations must be prepared to receive tabular information from agents that corresponds to entries not currently in use. Proper interpretation of such entries requires examination of the relevant inetNetToMediaType object.

The 'dynamic(3)' type indicates that the IP address to

physical addresses mapping has been dynamically resolved using e.g. IPv4 ARP or the IPv6 Neighbor Discovery protocol. The 'static(4)' type indicates that the mapping has been statically configured. Both of these refer to entries that provide mappings for other entities addresses.

The 'local(5)' type indicates that the mapping is provided for an entity's own interface address.

As the entries in this table are typically not persistent when this object is written the entity SHOULD NOT save the change to non-volatile storage."

```
::= { inetNetToMediaEntry 5 }
```

inetNetToMediaState OBJECT-TYPE

```
SYNTAX      INTEGER {
                reachable(1), -- confirmed reachability
                stale(2),      -- unconfirmed reachability
                delay(3),      -- waiting for reachability
                                -- confirmation before entering
                                -- the probe state
                probe(4),      -- actively probing
                invalid(5),    -- an invalidated mapping
                unknown(6),    -- state can not be determined
                                -- for some reason.
                incomplete(7) -- address resolution is being performed.
            }
```

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The Neighbor Unreachability Detection [[4](#)] state for the interface when the address mapping in this entry is used. If Neighbor Unreachability Detection is not in use (e.g. for IPv4), this object is always unknown(6)."

REFERENCE "[RFC2461](#)"

```
::= { inetNetToMediaEntry 6 }
```

```
--
-- The IPv6 Scope Identifier Table.
--
```



```
-- Open Issues:
--   Should there be associated objects to provide a scope description,
--   similar to ipMRouteScopeNameString?

-- XXX ScopeIdentifier TC should move to INET-ADDRESS-MIB
ScopeIdentifier ::= TEXTUAL-CONVENTION
    STATUS      current
    DESCRIPTION
        "A Scope Identifier identifies an instance of a specific
        scope.

        The scope identifier MUST disambiguate identical address
        values. For link-local addresses, the scope identifier will
        typically be the interface index (ifIndex as defined in the
        IF-MIB) of the interface on which the address is configured.

        The scope identifier may contain the special value 0 which
        refers to the default scope. The default scope may be used
        in cases where the valid scope identifier is not known
        (e.g., a management application needs to write a site-local
        InetAddressIPv6 address without knowing the site identifier
        value). The default scope SHOULD NOT be used as an easy way
        out in cases where the scope identifier for a non-global
        IPv6 address is known."
    SYNTAX      Unsigned32

ipv6ScopeIdTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Ipv6ScopeIdEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The table used to describe IPv6 unicast and multicast scope
        zones."
    ::= { ip 30 }

ipv6ScopeIdEntry OBJECT-TYPE
    SYNTAX      Ipv6ScopeIdEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Each entry contains the list of scope identifiers on a given
        interface."
    INDEX { ipv6ScopeIdIfIndex }
    ::= { ipv6ScopeIdTable 1 }

Ipv6ScopeIdEntry ::= SEQUENCE {
    ipv6ScopeIdIfIndex      InterfaceIndex,
    ipv6ScopeIdLinkLocal    ScopeIdentifier,
```



```
    ipv6ScopeIdSubnetLocal    ScopeIdentifier,  
    ipv6ScopeIdAdminLocal    ScopeIdentifier,  
    ipv6ScopeIdSiteLocal     ScopeIdentifier,  
    ipv6ScopeId6             ScopeIdentifier,  
    ipv6ScopeId7             ScopeIdentifier,  
    ipv6ScopeIdOrganizationLocal ScopeIdentifier,  
    ipv6ScopeId9             ScopeIdentifier,  
    ipv6ScopeIdA             ScopeIdentifier,  
    ipv6ScopeIdB             ScopeIdentifier,  
    ipv6ScopeIdC             ScopeIdentifier,  
    ipv6ScopeIdD             ScopeIdentifier  
}
```

ipv6ScopeIdIfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The index value which uniquely identifies the interface to which these scopes belong. The interface identified by a particular value of this index is the same interface as identified by the same value of the IF-MIB's ifIndex."

::= { ipv6ScopeIdEntry 1 }

ipv6ScopeIdLinkLocal OBJECT-TYPE

SYNTAX ScopeIdentifier

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The Scope Identifier for the link-local scope on this interface."

::= { ipv6ScopeIdEntry 2 }

ipv6ScopeIdSubnetLocal OBJECT-TYPE

SYNTAX ScopeIdentifier

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The Scope Identifier for the subnet-local scope on this interface."

::= { ipv6ScopeIdEntry 3 }

ipv6ScopeIdAdminLocal OBJECT-TYPE

SYNTAX ScopeIdentifier

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The Scope Identifier for the admin-local scope on this interface."


```
        interface."  
 ::= { ipv6ScopeIdEntry 4 }
```

ipv6ScopeIdSiteLocal OBJECT-TYPE

```
SYNTAX      ScopeIdentifier  
MAX-ACCESS  read-only  
STATUS      current  
DESCRIPTION  
    "The Scope Identifier for the site-local scope on this  
    interface."  
 ::= { ipv6ScopeIdEntry 5 }
```

ipv6ScopeId6 OBJECT-TYPE

```
SYNTAX      ScopeIdentifier  
MAX-ACCESS  read-only  
STATUS      current  
DESCRIPTION  
    "The Scope Identifier for scope 6 on this interface."  
 ::= { ipv6ScopeIdEntry 6 }
```

ipv6ScopeId7 OBJECT-TYPE

```
SYNTAX      ScopeIdentifier  
MAX-ACCESS  read-only  
STATUS      current  
DESCRIPTION  
    "The Scope Identifier for scope 7 on this interface."  
 ::= { ipv6ScopeIdEntry 7 }
```

ipv6ScopeIdOrganizationLocal OBJECT-TYPE

```
SYNTAX      ScopeIdentifier  
MAX-ACCESS  read-only  
STATUS      current  
DESCRIPTION  
    "The Scope Identifier for the orgainzation-local scope on  
    this interface."  
 ::= { ipv6ScopeIdEntry 8 }
```

ipv6ScopeId9 OBJECT-TYPE

```
SYNTAX      ScopeIdentifier  
MAX-ACCESS  read-only  
STATUS      current  
DESCRIPTION  
    "The Scope Identifier for scope 9 on this interface."  
 ::= { ipv6ScopeIdEntry 9 }
```

ipv6ScopeIdA OBJECT-TYPE

```
SYNTAX      ScopeIdentifier  
MAX-ACCESS  read-only
```



```
STATUS      current
DESCRIPTION
    "The Scope Identifier for scope A on this interface."
 ::= { ipv6ScopeIdEntry 10 }
```

ipv6ScopeIdB OBJECT-TYPE

```
SYNTAX      ScopeIdentifier
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The Scope Identifier for scope B on this interface."
 ::= { ipv6ScopeIdEntry 11 }
```

ipv6ScopeIdC OBJECT-TYPE

```
SYNTAX      ScopeIdentifier
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The Scope Identifier for scope C on this interface."
 ::= { ipv6ScopeIdEntry 12 }
```

ipv6ScopeIdD OBJECT-TYPE

```
SYNTAX      ScopeIdentifier
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The Scope Identifier for scope D on this interface."
 ::= { ipv6ScopeIdEntry 13 }
```

```
--
-- The Default Router Table
-- This table simply lists the default routers for more information
-- about routing tables see the routing MIBs
--
```

ipDefaultRouterTable OBJECT-TYPE

```
SYNTAX      SEQUENCE OF IpDefaultRouterEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "The table used to describe the default routers known to this
     entity."
 ::= { ip 32 }
```

ipDefaultRouterEntry OBJECT-TYPE

```
SYNTAX      IpDefaultRouterEntry
```


MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "Each entry contains information about a default router known
 to this entity."
INDEX {ipDefaultRouterAFType, ipDefaultRouterAddress}
::= { ipDefaultRouterTable 1 }

IpDefaultRouterEntry ::= SEQUENCE {
 ipDefaultRouterAFType InetAddressType,
 ipDefaultRouterAddress InetAddress,
 ipDefaultRouterIfIndex InterfaceIndex,
 ipDefaultRouterLifetime Unsigned32,
 ipDefaultRouterPreference INTEGER
}

ipDefaultRouterAFType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "The address family for this row."
::= { ipDefaultRouterEntry 1 }

ipDefaultRouterAddress OBJECT-TYPE
SYNTAX InetAddress (SIZE(0..36))
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "The IP address of the default router represented by this
 row."
::= { ipDefaultRouterEntry 2 }

ipDefaultRouterIfIndex OBJECT-TYPE
SYNTAX InterfaceIndex
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "The index value which uniquely identifies the interface by
 which the router can be reached. The interface identified
 by a particular value of this index is the same interface as
 identified by the same value of the IF-MIB's ifIndex."
::= { ipDefaultRouterEntry 3 }

ipDefaultRouterLifetime OBJECT-TYPE
SYNTAX Unsigned32 (0..65535)
UNITS "seconds"
MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The length of time, in seconds, that this router will remain useful as a default router. A value of zero indicates that it is no longer useful as a default router. It is left to the implementor of the MIB as to whether a router with a lifetime of zero is removed from the list.

For IPv6 this value should be extracted from the router advertisement messages. "

REFERENCE "For IPv6 [RFC2462](#) sections [4.2](#) and [6.3.4](#)"

::= { ipDefaultRouterEntry 4 }

ipDefaultRouterPreference OBJECT-TYPE

SYNTAX INTEGER {
High (1),
Reserved (2),
Low (3)
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"An indication of preference given to this router as a default router. Treating the value as a 2 bit signed integer allows for simple arithmetic comparisons.

For IPv4 routers or IPv6 routers that are not using the updated router advertisement format this object is set to medium (0)."

REFERENCE "[draft-ietf-ipv6-router-selection-02.txt](#), [section 2.1](#)"

::= { ipDefaultRouterEntry 5 }

--

-- Configuration information for constructing router advertisements

--

ipv6RouterAdvertTable OBJECT-TYPE

SYNTAX SEQUENCE OF Ipv6RouterAdvertEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The table containing information used to construct router advertisements."

::= { ip 33 }

ipv6RouterAdvertEntry OBJECT-TYPE

SYNTAX Ipv6RouterAdvertEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry containing information used to construct router advertisements.

Information in this table is persistent and when this object is written the entity SHOULD save the change to non-volatile storage."

INDEX { ipv6RouterAdvertIfIndex }

::= { ipv6RouterAdvertTable 1 }

```
Ipv6RouterAdvertEntry ::= SEQUENCE {  
    ipv6RouterAdvertIfIndex      InterfaceIndex,  
    ipv6RouterAdvertSendAdverts  TruthValue,  
    ipv6RouterAdvertMaxInterval  Unsigned32,  
    ipv6RouterAdvertMinInterval  Unsigned32,  
    ipv6RouterAdvertManagedFlag TruthValue,  
    ipv6RouterAdvertOtherConfigFlag TruthValue,  
    ipv6RouterAdvertLinkMTU      Unsigned32,  
    ipv6RouterAdvertReachableTime Unsigned32,  
    ipv6RouterAdvertRetransmitTime Unsigned32,  
    ipv6RouterAdvertCurHopLimit  Unsigned32,  
    ipv6RouterAdvertDefaultLifetime Unsigned32  
}
```

ipv6RouterAdvertIfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The index value which uniquely identifies the interface on which router advertisements constructed with this information will be transmitted. The interface identified by a particular value of this index is the same interface as identified by the same value of the IF-MIB's ifIndex."

::= { ipv6RouterAdvertEntry 1 }

ipv6RouterAdvertSendAdverts OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"A flag indicating whether or not the router sends periodic router advertisements and responds to router solicitations on this interface."

REFERENCE "[RFC2461 Section 6.2.1](#)"

DEFVAL { false }


```
::= { ipv6RouterAdvertEntry 2 }
```

ipv6RouterAdvertMaxInterval OBJECT-TYPE

SYNTAX Unsigned32 (4..1800)

UNITS "seconds"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum time allowed between sending unsolicited router advertisements from this interface."

REFERENCE "[RFC2461 Section 6.2.1](#)"

DEFVAL { 600 }

```
::= { ipv6RouterAdvertEntry 3 }
```

ipv6RouterAdvertMinInterval OBJECT-TYPE

SYNTAX Unsigned32 (3..1350)

UNITS "seconds"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The minimum time allowed between sending unsolicited router advertisements from this interface."

The default is 0.33 * ipv6RouterAdvertMaxInterval."

REFERENCE "[RFC2461 Section 6.2.1](#)"

```
::= { ipv6RouterAdvertEntry 4 }
```

ipv6RouterAdvertManagedFlag OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The true/false value to be placed into the 'managed address configuration' flag field in router advertisements sent from this interface."

REFERENCE "[RFC2461 Section 6.2.1](#)"

DEFVAL { FALSE }

```
::= { ipv6RouterAdvertEntry 5 }
```

ipv6RouterAdvertOtherConfigFlag OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The true/false value to be placed into the 'other stateful configuration' flag field in router advertisements sent from this interface."

REFERENCE "[RFC2461 Section 6.2.1](#)"


```
DEFVAL { FALSE }
::= { ipv6RouterAdvertEntry 6 }
```

ipv6RouterAdvertLinkMTU OBJECT-TYPE

```
SYNTAX      Unsigned32
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
```

"The value to be placed in MTU options sent by the router on this interface.

A value of zero indicates that no MTU options are sent."

REFERENCE ["RFC2461 Section 6.2.1"](#)

```
DEFVAL { 0 }
::= { ipv6RouterAdvertEntry 7 }
```

ipv6RouterAdvertReachableTime OBJECT-TYPE

```
SYNTAX      Unsigned32 (0..3600000)
UNITS "milliseconds"
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
```

"The value to be placed in the reachable time field in router advertisement messages sent from this interface.

A value of zero in the router advertisement indicates that the advertisement isn't specifying a value for reachable time."

REFERENCE ["RFC2461 Section 6.2.1"](#)

```
DEFVAL { 0 }
::= { ipv6RouterAdvertEntry 8 }
```

ipv6RouterAdvertRetransmitTime OBJECT-TYPE

```
SYNTAX      Unsigned32
UNITS "milliseconds"
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
```

"The value to be placed in the retransmit timer field in router advertisements sent from this interface.

A value of zero in the router advertisement indicates that the advertisement isn't specifying a value for retrans time."

REFERENCE ["RFC2461 Section 6.2.1"](#)

```
DEFVAL { 0 }
::= { ipv6RouterAdvertEntry 9 }
```


ipv6RouterAdvertCurHopLimit OBJECT-TYPE

SYNTAX Unsigned32 (0..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The default value to be placed in the current hop limit field in router advertisements sent from this interface. The value should be set to the current diameter of the Internet.

A value of zero in the router advertisement indicates that the advertisement isn't specifying a value for curHopLimit.

The default should be set to the value specified in the "Assigned Numbers" RFC that was in effect at the time of implementation."

REFERENCE "[RFC2461 Section 6.2.1](#)"

::= { ipv6RouterAdvertEntry 10 }

ipv6RouterAdvertDefaultLifetime OBJECT-TYPE

SYNTAX Unsigned32 (0..65535)

UNITS "seconds"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The value to be placed in the router lifetime field of router advertisements sent from this interface. This value MUST be either 0 or between ipv6RouterAdvertMaxInterval and 9000 seconds.

A value of zero indicates that the router is not to be used as a default router.

The default is 3 * ipv6RouterAdvertMaxInterval."

REFERENCE "[RFC2461 Section 6.2.1](#)"

::= { ipv6RouterAdvertEntry 11 }

--

-- ICMP section

--

icmp OBJECT IDENTIFIER ::= { mib-2 5 }

--

-- ICMP non-message-specific counters

--

inetIcmpTable OBJECT-TYPE

SYNTAX SEQUENCE OF InetIcmpEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The table of generic system-wide ICMP counters."

::= { icmp 27 }

inetIcmpEntry OBJECT-TYPE

SYNTAX InetIcmpEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row in the inetIcmpTable."

INDEX { inetIcmpAFType }

::= { inetIcmpTable 1 }

InetIcmpEntry ::= SEQUENCE {

inetIcmpAFType InetAddressType,

inetIcmpInMsgs Counter32,

inetIcmpInErrors Counter32,

inetIcmpOutMsgs Counter32,

inetIcmpOutErrors Counter32

}

inetIcmpAFType OBJECT-TYPE

SYNTAX InetAddressType

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The IP address family of the statistics."

::= { inetIcmpEntry 1 }

-- inetIcmpIfIndex

-- was ::= { inetIcmpEntry 2 }

-- XXX renumber?

inetIcmpInMsgs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of ICMP messages which the entity received.

Note that this counter includes all those counted by

inetIcmpInErrors."

::= { inetIcmpEntry 3 }

inetIcmpInErrors OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of ICMP messages which the entity received but determined as having ICMP-specific errors (bad ICMP checksums, bad length, etc.)."

::= { inetIcmpEntry 4 }

inetIcmpOutMsgs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of ICMP messages which the entity received. Note that this counter includes all those counted by inetIcmpOutErrors."

::= { inetIcmpEntry 5 }

inetIcmpOutErrors OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of ICMP messages which this entity did not send due to problems discovered within ICMP such as a lack of buffers. This value should not include errors discovered outside the ICMP layer such as the inability of IP to route the resultant datagram. In some implementations there may be no types of error which contribute to this counter's value."

::= { inetIcmpEntry 6 }

--

-- per-AF, per-message type ICMP counters

--

inetIcmpMsgTable OBJECT-TYPE

SYNTAX SEQUENCE OF InetIcmpMsgEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The table of system-wide per-message type ICMP counters."

::= { icmp 28 }

inetIcmpMsgEntry OBJECT-TYPE

SYNTAX InetIcmpMsgEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row in the inetIcmpMsgTable.

The system should track each ICMP type value, even if that
ICMP type is not supported by the system."

INDEX { inetIcmpMsgAftype, inetIcmpMsgType }

::= { inetIcmpMsgTable 1 }

InetIcmpMsgEntry ::= SEQUENCE {

inetIcmpMsgAftype InetAddressType,

inetIcmpMsgType Integer32,

inetIcmpMsgInPkts Counter32,

inetIcmpMsgOutPkts Counter32

}

inetIcmpMsgAftype OBJECT-TYPE

SYNTAX InetAddressType

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The IP address family of the statistics."

::= { inetIcmpMsgEntry 1 }

-- inetIcmpMsgIfIndex

-- was ::= { inetIcmpMsgEntry 2 }

-- XXX renumber?

inetIcmpMsgType OBJECT-TYPE

SYNTAX Integer32 (0..255)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The ICMP type field of the message type being counted by
this row.

Note that ICMP message types are scoped by the address
family in use."

REFERENCE "http://www.iana.org/assignments/icmp-parameters and
<http://www.iana.org/assignments/icmpv6-parameters>"

::= { inetIcmpMsgEntry 3 }

-- Removed

-- was ::= { inetIcmpMsgEntry 4 }

-- XXX Should we be renumbering?

inetIcmpMsgInPkts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of input packets for this AF and type."

::= { inetIcmpMsgEntry 5 }

inetIcmpMsgOutPkts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of output packets for this AF and type."

::= { inetIcmpMsgEntry 6 }

-- XXX

-- To do: move current conformance information here.

--

-- Deprecated objects

--

ipInReceives OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The total number of input datagrams received from
interfaces, including those received in error."

::= { ip 3 }

ipInHdrErrors OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of input datagrams discarded due to errors in
their IPv4 headers, including bad checksums, version number
mismatch, other format errors, time-to-live exceeded, errors
discovered in processing their IPv4 options, etc."

::= { ip 4 }

ipInAddrErrors OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION

"The number of input datagrams discarded because the IPv4 address in their IPv4 header's destination field was not a valid address to be received at this entity. This count includes invalid addresses (e.g., 0.0.0.0) and addresses of unsupported Classes (e.g., Class E). For entities which are not IPv4 routers and therefore do not forward datagrams, this counter includes datagrams discarded because the destination address was not a local address."

::= { ip 5 }

ipForwDatagrams OBJECT-TYPE

SYNTAX Counter32
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION

"The number of input datagrams for which this entity was not their final IPv4 destination, as a result of which an attempt was made to find a route to forward them to that final destination. In entities which do not act as IPv4 routers, this counter will include only those packets which were Source-Routed via this entity, and the Source-Route option processing was successful."

::= { ip 6 }

ipInUnknownProtos OBJECT-TYPE

SYNTAX Counter32
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION

"The number of locally-addressed datagrams received successfully but discarded because of an unknown or unsupported protocol."

::= { ip 7 }

ipInDiscards OBJECT-TYPE

SYNTAX Counter32
MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION

"The number of input IPv4 datagrams for which no problems were encountered to prevent their continued processing, but which were discarded (e.g., for lack of buffer space). Note that this counter does not include any datagrams discarded while awaiting re-assembly."

::= { ip 8 }

ipInDelivers OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The total number of input datagrams successfully delivered to IPv4 user-protocols (including ICMP)."

::= { ip 9 }

ipOutRequests OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The total number of IPv4 datagrams which local IPv4 user protocols (including ICMP) supplied to IPv4 in requests for transmission. Note that this counter does not include any datagrams counted in ipForwDatagrams."

::= { ip 10 }

ipOutDiscards OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of output IPv4 datagrams for which no problem was encountered to prevent their transmission to their destination, but which were discarded (e.g., for lack of buffer space). Note that this counter would include datagrams counted in ipForwDatagrams if any such packets met this (discretionary) discard criterion."

::= { ip 11 }

ipOutNoRoutes OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of IPv4 datagrams discarded because no route could be found to transmit them to their destination. Note that this counter includes any packets counted in ipForwDatagrams which meet this 'no-route' criterion. Note that this includes any datagrams which a host cannot route because all of its default routers are down."

::= { ip 12 }

ipReasmReqds OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only
STATUS deprecated
DESCRIPTION
 "The number of IPv4 fragments received which needed to be
 reassembled at this entity."
 ::= { ip 14 }

ipReasmOKs OBJECT-TYPE
 SYNTAX Counter32
 MAX-ACCESS read-only
 STATUS deprecated
 DESCRIPTION
 "The number of IPv4 datagrams successfully re-assembled."
 ::= { ip 15 }

ipReasmFails OBJECT-TYPE
 SYNTAX Counter32
 MAX-ACCESS read-only
 STATUS deprecated
 DESCRIPTION
 "The number of failures detected by the IPv4 re-assembly
 algorithm (for whatever reason: timed out, errors, etc).
 Note that this is not necessarily a count of discarded IPv4
 fragments since some algorithms (notably the algorithm in
 [RFC 815](#)) can lose track of the number of fragments by
 combining them as they are received."
 ::= { ip 16 }

ipFragOKs OBJECT-TYPE
 SYNTAX Counter32
 MAX-ACCESS read-only
 STATUS deprecated
 DESCRIPTION
 "The number of IPv4 datagrams that have been successfully
 fragmented at this entity."
 ::= { ip 17 }

ipFragFails OBJECT-TYPE
 SYNTAX Counter32
 MAX-ACCESS read-only
 STATUS deprecated
 DESCRIPTION
 "The number of IPv4 datagrams that have been discarded
 because they needed to be fragmented at this entity but
 could not be, e.g., because their Don't Fragment flag was
 set."
 ::= { ip 18 }

ipFragCreates OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of IPv4 datagram fragments that have been generated as a result of fragmentation at this entity."

::= { ip 19 }

ipRoutingDiscards OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of routing entries which were chosen to be discarded even though they are valid. One possible reason for discarding such an entry could be to free-up buffer space for other routing entries."

::= { ip 23 }

-- the deprecated IPv4 address table

ipAddrTable OBJECT-TYPE

SYNTAX SEQUENCE OF IpAddrEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"The table of addressing information relevant to this entity's IPv4 addresses."

::= { ip 20 }

ipAddrEntry OBJECT-TYPE

SYNTAX IpAddrEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"The addressing information for one of this entity's IPv4 addresses."

INDEX { ipAdEntAddr }

::= { ipAddrTable 1 }

IpAddrEntry ::= SEQUENCE {

ipAdEntAddr IpAddress,

ipAdEntIfIndex INTEGER,

ipAdEntNetMask IpAddress,

ipAdEntBcastAddr INTEGER,

ipAdEntReasmMaxSize INTEGER

}

ipAdEntAddr OBJECT-TYPE

SYNTAX IpAddress

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The IPv4 address to which this entry's addressing information pertains."

::= { ipAddrEntry 1 }

ipAdEntIfIndex OBJECT-TYPE

SYNTAX INTEGER (1..2147483647)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The index value which uniquely identifies the interface to which this entry is applicable. The interface identified by a particular value of this index is the same interface as identified by the same value of the IF-MIB's ifIndex."

::= { ipAddrEntry 2 }

ipAdEntNetMask OBJECT-TYPE

SYNTAX IpAddress

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The subnet mask associated with the IPv4 address of this entry. The value of the mask is an IPv4 address with all the network bits set to 1 and all the hosts bits set to 0."

::= { ipAddrEntry 3 }

ipAdEntBcastAddr OBJECT-TYPE

SYNTAX INTEGER (0..1)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The value of the least-significant bit in the IPv4 broadcast address used for sending datagrams on the (logical) interface associated with the IPv4 address of this entry. For example, when the Internet standard all-ones broadcast address is used, the value will be 1. This value applies to both the subnet and network broadcasts addresses used by the entity on this (logical) interface."

::= { ipAddrEntry 4 }

ipAdEntReasmMaxSize OBJECT-TYPE

SYNTAX INTEGER (0..65535)

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The size of the largest IPv4 datagram which this entity can re-assemble from incoming IPv4 fragmented datagrams received on this interface."

::= { ipAddrEntry 5 }

-- the deprecated IPv4 Address Translation table

-- The Address Translation tables contain the IpAddress to
-- "physical" address equivalences. Some interfaces do not
-- use translation tables for determining address
-- equivalences (e.g., DDN-X.25 has an algorithmic method);
-- if all interfaces are of this type, then the Address
-- Translation table is empty, i.e., has zero entries.

ipNetToMediaTable OBJECT-TYPE

SYNTAX SEQUENCE OF IpNetToMediaEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"The IPv4 Address Translation table used for mapping from IPv4 addresses to physical addresses."

::= { ip 22 }

ipNetToMediaEntry OBJECT-TYPE

SYNTAX IpNetToMediaEntry

MAX-ACCESS not-accessible

STATUS deprecated

DESCRIPTION

"Each entry contains one IpAddress to 'physical' address equivalence."

INDEX { ipNetToMediaIfIndex,
ipNetToMediaNetAddress }

::= { ipNetToMediaTable 1 }

IpNetToMediaEntry ::= SEQUENCE {

ipNetToMediaIfIndex INTEGER,
ipNetToMediaPhysAddress PhysAddress,
ipNetToMediaNetAddress IpAddress,
ipNetToMediaType INTEGER

}

ipNetToMediaIfIndex OBJECT-TYPE

SYNTAX INTEGER (1..2147483647)

MAX-ACCESS read-create

STATUS deprecated

DESCRIPTION

"The interface on which this entry's equivalence is effective. The interface identified by a particular value of this index is the same interface as identified by the same value of the IF-MIB's ifIndex."

::= { ipNetToMediaEntry 1 }

ipNetToMediaPhysAddress OBJECT-TYPE

SYNTAX PhysAddress

MAX-ACCESS read-create

STATUS deprecated

DESCRIPTION

"The media-dependent `physical' address. This object should return 0 when this entry is in the "incomplete" state.

As the entries in this table are typically not persistent when this object is written the entity should not save the change to non-volatile storage. Note: a stronger requirement is not used because this object was previously defined."

::= { ipNetToMediaEntry 2 }

ipNetToMediaNetAddress OBJECT-TYPE

SYNTAX IpAddress

MAX-ACCESS read-create

STATUS deprecated

DESCRIPTION

"The IpAddress corresponding to the media-dependent `physical' address."

::= { ipNetToMediaEntry 3 }

ipNetToMediaType OBJECT-TYPE

SYNTAX INTEGER {
 other(1), -- none of the following
 invalid(2), -- an invalidated mapping
 dynamic(3),
 static(4)
}

MAX-ACCESS read-create

STATUS deprecated

DESCRIPTION

"The type of mapping.

Setting this object to the value invalid(2) has the effect of invalidating the corresponding entry in the ipNetToMediaTable. That is, it effectively disassociates the interface identified with said entry from the mapping

identified with said entry. It is an implementation-specific matter as to whether the agent removes an invalidated entry from the table. Accordingly, management stations must be prepared to receive tabular information from agents that corresponds to entries not currently in use. Proper interpretation of such entries requires examination of the relevant `ipNetToMediaType` object.

As the entries in this table are typically not persistent when this object is written the entity should not save the change to non-volatile storage. Note: a stronger requirement is not used because this object was previously defined."

```
::= { ipNetToMediaEntry 4 }
```

-- the deprecated ICMP group

`icmpInMsgs` OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The total number of ICMP messages which the entity received.

Note that this counter includes all those counted by

`icmpInErrors`."

```
::= { icmp 1 }
```

`icmpInErrors` OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP messages which the entity received but determined as having ICMP-specific errors (bad ICMP checksums, bad length, etc.)."

```
::= { icmp 2 }
```

`icmpInDestUnreachs` OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Destination Unreachable messages received."

```
::= { icmp 3 }
```


icmpInTimeExcds OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Time Exceeded messages received."

::= { icmp 4 }

icmpInParmProbs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Parameter Problem messages received."

::= { icmp 5 }

icmpInSrcQuenchs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Source Quench messages received."

::= { icmp 6 }

icmpInRedirects OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Redirect messages received."

::= { icmp 7 }

icmpInEchos OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Echo (request) messages received."

::= { icmp 8 }

icmpInEchoReps OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Echo Reply messages received."

::= { icmp 9 }

icmpInTimestamps OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Timestamp (request) messages received."

::= { icmp 10 }

icmpInTimestampReps OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Timestamp Reply messages received."

::= { icmp 11 }

icmpInAddrMasks OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Address Mask Request messages received."

::= { icmp 12 }

icmpInAddrMaskReps OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Address Mask Reply messages received."

::= { icmp 13 }

icmpOutMsgs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The total number of ICMP messages which this entity attempted to send. Note that this counter includes all those counted by icmpOutErrors."

::= { icmp 14 }

icmpOutErrors OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP messages which this entity did not send"

due to problems discovered within ICMP such as a lack of buffers. This value should not include errors discovered outside the ICMP layer such as the inability of IP to route the resultant datagram. In some implementations there may be no types of error which contribute to this counter's value."

::= { icmp 15 }

icmpOutDestUnreachs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Destination Unreachable messages sent."

::= { icmp 16 }

icmpOutTimeExcds OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Time Exceeded messages sent."

::= { icmp 17 }

icmpOutParmProbs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Parameter Problem messages sent."

::= { icmp 18 }

icmpOutSrcQuenchs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Source Quench messages sent."

::= { icmp 19 }

icmpOutRedirects OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Redirect messages sent. For a host, this object will always be zero, since hosts do not send redirects."


```
::= { icmp 20 }
```

icmpOutEchos OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Echo (request) messages sent."

```
::= { icmp 21 }
```

icmpOutEchoReps OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Echo Reply messages sent."

```
::= { icmp 22 }
```

icmpOutTimestamps OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Timestamp (request) messages sent."

```
::= { icmp 23 }
```

icmpOutTimestampReps OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Timestamp Reply messages sent."

```
::= { icmp 24 }
```

icmpOutAddrMasks OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Address Mask Request messages sent."

```
::= { icmp 25 }
```

icmpOutAddrMaskReps OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS deprecated

DESCRIPTION

"The number of ICMP Address Mask Reply messages sent."


```
 ::= { icmp 26 }

-- conformance information

ipMIBConformance OBJECT IDENTIFIER ::= { ipMIB 2 }

ipMIBCompliances OBJECT IDENTIFIER ::= { ipMIBConformance 1 }
ipMIBGroups      OBJECT IDENTIFIER ::= { ipMIBConformance 2 }

ipv6Conformance OBJECT IDENTIFIER ::= { ipv6MIB 3 }

ipv6Compliances OBJECT IDENTIFIER ::= { ipv6Conformance 1 }
ipv6Groups      OBJECT IDENTIFIER ::= { ipv6Conformance 2 }

-- ipv6IcmpConformance OBJECT IDENTIFIER ::= { ipv6IcmpMIB 2 }
--
-- ipv6IcmpCompliances OBJECT IDENTIFIER ::= { ipv6IcmpConformance 1 }
-- ipv6IcmpGroups      OBJECT IDENTIFIER ::= { ipv6IcmpConformance 2 }

-- compliance statements

ipMIBCompliance MODULE-COMPLIANCE
    STATUS      deprecated
    DESCRIPTION
        "The compliance statement for systems which implement only
        IPv4.  For version-independence, this compliance statement
        is deprecated in favor of ipMIBCompliance2."
    MODULE -- this module
        MANDATORY-GROUPS { ipGroup,
                           icmpGroup }
    ::= { ipMIBCompliances 1 }

-- ipv6Compliance MODULE-COMPLIANCE
-- .ST c
-- .(D
-- "The compliance statement for systems which
-- implement ipv6 MIB."
-- .)D
--
-- MODULE -- -- this module
--
-- MANDATORY-GROUPS { ipv6GeneralGroup,
--                    ipv6NotificationGroup }
--
-- OBJECT      ipv6Forwarding
-- MIN-ACCESS  read-only
-- .(D
-- "An agent is not required to provide write
-- access to this object"
```



```
-- .)D
--          OBJECT    ipv6DefaultHopLimit
--          MIN-ACCESS  read-only
-- .(D
-- "An agent is not required to provide write
-- access to this object"
-- .)D
--          OBJECT    ipv6IfDescr
--          MIN-ACCESS  read-only
-- .(D
-- "An agent is not required to provide write
-- access to this object"
-- .)D
--          OBJECT    ipv6IfIdentifier
--          MIN-ACCESS  read-only
-- .(D
-- "An agent is not required to provide write
-- access to this object"
-- .)D
--          OBJECT    ipv6IfIdentifierLength
--          MIN-ACCESS  read-only
-- .(D
-- "An agent is not required to provide write
-- access to this object"
-- .)D
--          OBJECT    ipv6IfAdminStatus
--          MIN-ACCESS  read-only
-- .(D
-- "An agent is not required to provide write
-- access to this object"
-- .)D
--          OBJECT    ipv6RouteValid
--          MIN-ACCESS  read-only
-- .(D
-- "An agent is not required to provide write
-- access to this object"
-- .)D
--          OBJECT    ipv6NetToMediaValid
--          MIN-ACCESS  read-only
-- .(D
-- "An agent is not required to provide write
-- access to this object"
-- .)D
--          ::= { ipv6Compliances 1 }
```


-- units of conformance

ipGroup2 OBJECT-GROUP

OBJECTS { ipForwarding, ipDefaultTTL }

STATUS current

DESCRIPTION

"The group of IPv4-specific objects for basic management of
IPv4 entities."

::= { ipMIBGroups 3 }

ipIfStatsGroup OBJECT-GROUP

OBJECTS { ipIfStatsInReceives, ipIfStatsInHdrErrors,
ipIfStatsInTooBigErrors, ipIfStatsInNoRoutes,
ipIfStatsInAddrErrors,

ipIfStatsInUnknownProtos,

ipIfStatsInTruncatedPkts, ipIfStatsInDiscards,
ipIfStatsInDelivers,

ipIfStatsOutForwDatagrams,

ipIfStatsOutRequests, ipIfStatsOutDiscards,
ipIfStatsOutFragOKs, ipIfStatsOutFragFails,
ipIfStatsOutFragCreates, ipIfStatsReasmReqds,
ipIfStatsReasmOKs, ipIfStatsReasmFails,
ipIfStatsInMcastPkts, ipIfStatsOutMcastPkts,
ipIfStatsInOctets, ipIfStatsOutOctets,
ipIfStatsInBcastPkts, ipIfStatsOutBcastPkts,
ipIfStatsInMcastOctets,

ipIfStatsOutMcastOctets }

STATUS current

DESCRIPTION

"IP per-interface or per-system statistics."

::= { ipMIBGroups 4 }

-- XXX some HC statistics groups

ipv6ScopeGroup OBJECT-GROUP

OBJECTS { ipv6ScopeIdLinkLocal, ipv6ScopeIdSubnetLocal,
ipv6ScopeIdAdminLocal, ipv6ScopeIdSiteLocal,
ipv6ScopeId6, ipv6ScopeId7,
ipv6ScopeIdOrganizationLocal, ipv6ScopeId9,
ipv6ScopeIdA, ipv6ScopeIdB,
ipv6ScopeIdC, ipv6ScopeIdD }

STATUS current

DESCRIPTION

"The group of objects for managing IPv6 scope zones."

::= { ipMIBGroups 5 }

ipGroup OBJECT-GROUP

OBJECTS { ipForwarding, ipDefaultTTL, ipInReceives,

ipInHdrErrors, ipInAddrErrors,
ipForwDatagrams, ipInUnknownProtos,
ipInDiscards, ipInDelivers, ipOutRequests,

```
ipOutDiscards, ipOutNoRoutes,
ipReasmTimeout, ipReasmReqds, ipReasmOKs,
ipReasmFails, ipFragOKs,
ipFragFails, ipFragCreates,
ipAdEntAddr, ipAdEntIfIndex, ipAdEntNetMask,
ipAdEntBcastAddr, ipAdEntReasmMaxSize,
ipNetToMediaIfIndex, ipNetToMediaPhysAddress,
ipNetToMediaNetAddress, ipNetToMediaType,
ipRoutingDiscards }
```

STATUS deprecated

DESCRIPTION

"The ip group of objects providing for basic management of IP entities, exclusive of the management of IP routes."

::= { ipMIBGroups 1 }

icmpGroup OBJECT-GROUP

```
OBJECTS { icmpInMsgs, icmpInErrors,
icmpInDestUnreaches, icmpInTimeExcds,
icmpInParmProbs, icmpInSrcQuenches,
icmpInRedirects, icmpInEchos,
icmpInEchoReps, icmpInTimestamps,
icmpInTimestampReps, icmpInAddrMasks,
icmpInAddrMaskReps, icmpOutMsgs,
icmpOutErrors, icmpOutDestUnreaches,
icmpOutTimeExcds, icmpOutParmProbs,
icmpOutSrcQuenches, icmpOutRedirects,
icmpOutEchos, icmpOutEchoReps,
icmpOutTimestamps, icmpOutTimestampReps,
icmpOutAddrMasks, icmpOutAddrMaskReps }
```

STATUS deprecated

DESCRIPTION

"The icmp group of objects providing ICMP statistics."

::= { ipMIBGroups 2 }

ipv6GeneralGroup2 OBJECT-GROUP

```
OBJECTS { ipv6Forwarding,
ipv6DefaultHopLimit }
```

STATUS current

DESCRIPTION

"The IPv6 group of objects providing for basic management of IPv6 entities."

::= { ipv6Groups 3 }

END

6. Open Issues / To Do

The MIB has not been passed through a MIB checker yet - after we have some agreement on the objects I'll tidy up the MIB and have it checked.

Need descriptive intro text, or LOTS more stuff in DESCRIPTIONs.

Any other objects from ipv6IfTable that we need? When can ipv6IfPhysicalAddress be different than the interface's ifPhysAddress?

How to describe what stats are required, especially in the ICMP Msg table? All ICMP statistics to be required.

Need to update conformance info.

Are there other possible sources for ipAddressPrefixOrigin?

Should we include an object or objects to express the capability of an implementation with respect to items like extension headers for IPv6.

Are there other items that could be added to the ipv{4 6}IFTables such as (for v4) directed broadcast, proxy arp, header compression, broadcast address and arp cache timeout.

Should we add lastChange and/or Number objects to some or all tables.

I have chosen not to add objects for the following items from the IPv6 specifications: DupAddrDetectTransmits, managed flag, otherconfig flag and isRouter flag.

I have added a table for router advertisement configuration objects and made the objects read-write. Is the table useful? should the objects be something else (read-only or read-create)?

I have not added a table for the prefixes that will be added to router advertisements feeling that such information should be in the router mib area.

Note: more open issues / to do items scattered in comments in MIB.

7. Acknowledgements

This document contains objects modified from [RFC 1213](#) [2], [RFC 2011](#) [3], [RFC 2465](#) [6], and [RFC 2466](#) [7].

8. References

- [1] D. Plummer, "An Ethernet Address Resolution Protocol", [RFC0826](#), November 1982.
- [2] Rose, M. and K. McCloghrie, "Management Information Base for Network Management of TCP/IP-based internets", [RFC 1213](#), March 1991.
- [3] K. McCloghrie, "SNMPv2 Management Information Base for the Internet Protocol using SMIV2", [RFC 2011](#), November 1996.
- [4] Narten, T., E. Nordmark and W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)", [RFC 2461](#), December 1998.
- [5] Thomson, S., T. Narten, "IPv6 Stateless Address Autoconfiguration", [RFC 2462](#), December 1998.
- [6] Haskin, D. and S. Onishi, "Management Information Base for IP Version 6: Textual Conventions and General Group", [RFC 2465](#), December 1998.
- [7] Haskin, D. and S. Onishi, "Management Information Base for IP Version 6: ICMPv6 Group", [RFC 2466](#), December 1998.
- [8] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing SNMP Management Frameworks", [RFC 2571](#), April 1999.
- [9] Rose, M., and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", STD 16, [RFC 1155](#), May 1990.
- [10] Rose, M., and K. McCloghrie, "Concise MIB Definitions", STD 16, [RFC 1212](#), March 1991.
- [11] Rose, M., "A Convention for Defining Traps for use with the SNMP", [RFC 1215](#), March 1991.
- [12] 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.
- [13] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Textual Conventions for SMIV2", STD 58, [RFC 2579](#), April 1999.
- [14] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Conformance Statements for SMIV2", STD 58, RFC

2580, April 1999.

- [15] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol", STD 15, [RFC 1157](#), May 1990.
- [16] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Introduction to Community-based SNMPv2", [RFC 1901](#), January 1996.
- [17] 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.
- [18] Case, J., Harrington D., Presuhn R., and B. Wijnen, "Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)", [RFC 2572](#), April 1999.
- [19] Blumenthal, U., and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", [RFC 2574](#), April 1999.
- [20] 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.
- [21] Levi, D., Meyer, P., and B. Stewart, "SNMPv3 Applications", [RFC 2573](#), April 1999.
- [22] Wijnen, B., Presuhn, R., and K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)", [RFC 2575](#), April 1999.
- [23] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction to Version 3 of the Internet-standard Network Management Framework", [RFC 2570](#), April 1999.

9. Security Considerations

There are a number of management objects defined in this MIB that have a MAX-ACCESS clause of read-write and/or read-create. Such objects may be considered sensitive or vulnerable in some network environments. The support for SET operations in a non-secure environment without proper protection can have a negative effect on network operations.

There are a number of managed objects in this MIB that may contain sensitive information. These are:

-- not yet evaluated

It is thus important to control even GET access to these objects and possibly to even encrypt the values of these object when sending them over the network via SNMP. Not all versions of SNMP provide features for such a secure environment.

SNMPv1 by itself is not a secure environment. Even if the network itself is secure (for example by using IPSec), even then, there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB.

It is recommended that the implementers consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model [RFC 2574](#) [19] and the View-based Access Control Model [RFC 2575](#) [22] is recommended.

It is then a customer/user responsibility to ensure that the SNMP entity giving access to an instance of this MIB, is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

[10.](#) Editor's Contact Information

Shawn A. Routhier
Wind River
500 Wind River Way
Alameda, CA 94501
USA

Email: sar@epilogue.com

[11.](#) Authors

This document was written by the IPv6 MIB revision design team:

Bill Fenner, AT&T Labs -- Research
Email: fenner@research.att.com

Brian Haberman
Email: bkhabs@nc.rr.com

Shawn A. Routhier, Wind River
Email: sar@epilogue.com

Dave Thaler, Microsoft
Email: dthaler@windows.microsoft

This document updates parts of the MIBs from several other documents.
[RFC2011](#) is the previous update to the IP MIB. [RFC2465](#) and [RFC2466](#) are the first versions specifying IPv6 addresses and information.

[RFC2011](#):

Keith McCloghrie, Cisco Systems (Editor)

[RFC2465](#) and [RFC2466](#):

Dimitry Haskin, Bay Networks

Steve Onishi, Bay Networks

[12.](#) Acknowledgements

Reviews and other contributions were made by:

Dario Acornero, Cisco

Mike MacFaden, Riverstone

Juergen Schoenwalder, TU Braunschweig

Margaret Wasserman, Wind River

[13.](#) 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.

--S.x5ty08154=_/stovokor.epilogue.com--