

Definitions of Managed Objects  
for the Border Gateway Protocol (Version 3)

Status of this Memo

This memo is an extension to the SNMP MIB. This RFC specifies an IAB standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "IAB Official Protocol Standards" for the standardization state and status of this protocol. Distribution of this memo is unlimited.

1. Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in TCP/IP-based internets. In particular, it defines objects for managing the Border Gateway Protocol [11,12].

2. The Network Management Framework

The Internet-standard Network Management Framework consists of three components. They are:

RFC 1155 which defines the SMI, the mechanisms used for describing and naming objects for the purpose of management. RFC 1212 defines a more concise description mechanism, which is wholly consistent with the SMI.

RFC 1156 which defines MIB-I, the core set of managed objects for the Internet suite of protocols. RFC 1213, defines MIB-II, an evolution of MIB-I based on implementation experience and new operational requirements.

RFC 1157 which defines the SNMP, the protocol used for network access to managed objects.

The Framework permits new objects to be defined for the purpose of experimentation and evaluation.

### 3. Objects

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) [7] defined in the SMI. In particular, each object has a name, a syntax, and an encoding. The name is an object identifier, an administratively assigned name, which specifies an object type. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the OBJECT DESCRIPTOR, to also refer to the object type.

The syntax of an object type defines the abstract data structure corresponding to that object type. The ASN.1 language is used for this purpose. However, the SMI [3] purposely restricts the ASN.1 constructs which may be used. These restrictions are explicitly made for simplicity.

The encoding of an object type is simply how that object type is represented using the object type's syntax. Implicitly tied to the notion of an object type's syntax and encoding is how the object type is represented when being transmitted on the network.

The SMI specifies the use of the basic encoding rules of ASN.1 [8], subject to the additional requirements imposed by the SNMP.

#### 3.1. Format of Definitions

Section 5 contains contains the specification of all object types contained in this MIB module. The object types are defined using the conventions defined in the SMI, as amended by the extensions specified in [9,10].

### 4. Overview

These objects are used to control and manage a BGP [11,12] implementation.

The Border Gateway Protocol (BGP) is an inter-Autonomous System routing protocol. The primary function of a BGP speaking system is to exchange network reachability information with other BGP systems. This network reachability information includes information on the full path of Autonomous Systems that traffic must transit to reach these networks.

BGP runs over a reliable transport protocol. This eliminates the need to implement explicit update fragmentation, retransmission,

acknowledgement, and sequencing. Any authentication scheme used by the transport protocol may be used in addition to BGP's own authentication mechanisms.

The planned use of BGP in the Internet environment, including such issues as topology, the interaction between BGP and IGPs, and the enforcement of routing policy rules is presented in a companion document [12].

Apart from a few system variables, this MIB is broken into two tables: the BGP Peer Table and the BGP Received Path Attribute Table. The Peer Table reflects information about BGP peer connections, such as their state and current activity. The Received Path Attribute Table contains all attributes received from all peers before local routing policy has been applied. The actual attributes used in determining a route are a subset of the received attribute table.

## 5. Definitions

```
RFC1269-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    NetworkAddress, IpAddress, Counter
```

```
    FROM RFC1155-SMI
```

```
    mib-2
```

```
    FROM RFC1213-MIB
```

```
OBJECT-TYPE
```

```
    FROM RFC-1212
```

```
TRAP-TYPE
```

```
    FROM RFC-1215;
```

```
-- This MIB module uses the extended OBJECT-TYPE macro as
-- defined in [9], and the TRAP-TYPE macro as defined
-- in [10].
```

```
bgp      OBJECT IDENTIFIER ::= { mib-2 15 }
```

```
bgpVersion OBJECT-TYPE
```

```
    SYNTAX OCTET STRING
```

```
    ACCESS read-only
```

```
    STATUS mandatory
```

```
    DESCRIPTION
```

```
        "Vector of supported BGP protocol version
        numbers. Each peer negotiates the version from
        this vector. Versions are identified via the
        string of bits contained within this object.
        The first octet contains bits 0 to 7, the
        second octet contains bits 8 to 15, and so on,
```

with the most significant bit referring to the lowest bit number in the octet (e.g., the MSB of the first octet refers to bit 0). If a bit, i, is present and set, then the version (i+1) of the BGP is supported."

::= { bgp 1 }

bgpLocalAs OBJECT-TYPE

SYNTAX INTEGER (0..65535)

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The local autonomous system number."

::= { bgp 2 }

bgpPeerTable OBJECT-TYPE

SYNTAX SEQUENCE OF BgpPeerEntry

ACCESS not-accessible

STATUS mandatory

DESCRIPTION

"The bgp peer table."

::= { bgp 3 }

bgpIdentifier OBJECT-TYPE

SYNTAX IpAddress

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The BGP Identifier of local system."

::= { bgp 4 }

bgpPeerEntry OBJECT-TYPE

SYNTAX BgpPeerEntry

ACCESS not-accessible

STATUS mandatory

DESCRIPTION

"Information about a BGP peer connection."

INDEX

{ bgpPeerRemoteAddr }

::= { bgpPeerTable 1 }

BgpPeerEntry ::= SEQUENCE {

bgpPeerIdentifier

IpAddress,

bgpPeerState

INTEGER,

bgpPeerAdminStatus

INTEGER,

```
    bgpPeerNegotiatedVersion
        INTEGER,
    bgpPeerLocalAddr
        IpAddress,
    bgpPeerLocalPort
        INTEGER,
    bgpPeerRemoteAddr
        IpAddress,
    bgpPeerRemotePort
        INTEGER,
    bgpPeerRemoteAs
        INTEGER,
    bgpPeerInUpdates
        Counter,
    bgpPeerOutUpdates
        Counter,
    bgpPeerInTotalMessages
        Counter,
    bgpPeerOutTotalMessages
        Counter,
    bgpPeerLastError
        OCTET STRING
}

bgpPeerIdentifier OBJECT-TYPE
    SYNTAX IpAddress
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The BGP Identifier of this entry's BGP peer."
    ::= { bgpPeerEntry 1 }

bgpPeerState OBJECT-TYPE
    SYNTAX INTEGER {
        idle(1),
        connect(2),
        active(3),
        opensent(4),
        openconfirm(5),
        established(6)
    }
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The bgp peer connection state. "
    ::= { bgpPeerEntry 2 }
```

bgpPeerAdminStatus OBJECT-TYPE  
SYNTAX INTEGER  
ACCESS read-write  
STATUS mandatory  
DESCRIPTION  
    "The desired state of the BGP connection. A transition from 'stop' to 'start' will cause the BGP Start Event to be generated. A transition from 'start' to 'stop' will cause the BGP Stop Event to be generated. This parameter can be used to restart BGP peer connections. Care should be used in providing write access to this object without adequate authentication."  
 ::= { bgpPeerEntry 3 }

bgpPeerNegotiatedVersion OBJECT-TYPE  
SYNTAX INTEGER  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "The negotiated version of BGP running between the two peers. "  
 ::= { bgpPeerEntry 4 }

bgpPeerLocalAddr OBJECT-TYPE  
SYNTAX IpAddress  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "The local IP address of this entry's BGP connection."  
 ::= { bgpPeerEntry 5 }

bgpPeerLocalPort OBJECT-TYPE  
SYNTAX INTEGER (0..65535)  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "The local port for the TCP connection between the BGP peers."  
 ::= { bgpPeerEntry 6 }

bgpPeerRemoteAddr OBJECT-TYPE  
SYNTAX IpAddress  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION

```
        "The remote IP address of this entry's BGP
        peer."
 ::= { bgpPeerEntry 7 }

bgpPeerRemotePort OBJECT-TYPE
    SYNTAX INTEGER (0..65535)
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The remote port for the TCP connection between
        the BGP peers.  Note that the objects
        bgpLocalAddr, bgpLocalPort, bgpRemoteAddr and
        bgpRemotePort provide the appropriate reference
        to the standard MIB TCP connection table."
 ::= { bgpPeerEntry 8 }

bgpPeerRemoteAs OBJECT-TYPE
    SYNTAX INTEGER (0..65535)
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The remote autonomous system number."
 ::= { bgpPeerEntry 9 }

bgpPeerInUpdates OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of BGP UPDATE messages received on
        this connection.  This object should be
        initialized to zero when the connection is
        established."
 ::= { bgpPeerEntry 10 }

bgpPeerOutUpdates OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of BGP UPDATE messages received on
        this connection.  This object should be
        initialized to zero when the connection is
        established."
 ::= { bgpPeerEntry 11 }

bgpPeerInTotalMessages OBJECT-TYPE
    SYNTAX Counter
```

ACCESS read-only  
STATUS mandatory  
DESCRIPTION

"The total number of messages received from the remote peer on this connection. This object should be initialized to zero when the connection is established."

::= { bgpPeerEntry 12 }

bgpPeerOutTotalMessages OBJECT-TYPE

SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION

"The total number of messages transmitted to the remote peer on this connection. This object should be initialized to zero when the connection is established."

::= { bgpPeerEntry 13 }

bgpPeerLastError OBJECT-TYPE

SYNTAX OCTET STRING (SIZE (2))  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION

"The last error code and subcode seen by this peer on this connection. If no error has occurred, this field is zero. Otherwise, the first byte of this two byte OCTET STRING contains the error code; the second contains the subcode."

::= { bgpPeerEntry 14 }

bgpRcvdPathAttrTable OBJECT-TYPE

SYNTAX SEQUENCE OF BgpPathAttrEntry  
ACCESS not-accessible  
STATUS mandatory  
DESCRIPTION

"The BGP Received Path Attribute Table contains information about paths to destination networks received by all peers."

::= { bgp 5 }

bgpPathAttrEntry OBJECT-TYPE

SYNTAX BgpPathAttrEntry  
ACCESS not-accessible  
STATUS mandatory  
DESCRIPTION



```

        "Information about a path to a network."
INDEX
    { bgpPathAttrDestNetwork,
      bgpPathAttrPeer }
 ::= { bgpRcvdPathAttrTable 1 }

BgpPathAttrEntry ::= SEQUENCE {
    bgpPathAttrPeer
        IPAddress,
    bgpPathAttrDestNetwork
        IPAddress,
    bgpPathAttrOrigin
        INTEGER,
    bgpPathAttrASPath
        OCTET STRING,
    bgpPathAttrNextHop
        IPAddress,
    bgpPathAttrInterASMetric
        INTEGER
    }

bgpPathAttrPeer OBJECT-TYPE
    SYNTAX IPAddress
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The IP address of the peer where the path
        information
        was learned."
    ::= { bgpPathAttrEntry 1 }

bgpPathAttrDestNetwork OBJECT-TYPE
    SYNTAX IPAddress
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The address of the destination network."
    ::= { bgpPathAttrEntry 2 }

bgpPathAttrOrigin OBJECT-TYPE
    SYNTAX INTEGER {
        igp(1),-- networks are interior
        egp(2),-- networks learned via EGP
        incomplete(3) -- undetermined
    }
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION

```

"The ultimate origin of the path information."  
 ::= { bgpPathAttrEntry 3 }

bgpPathAttrASPath OBJECT-TYPE

SYNTAX OCTET STRING

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The set of ASs that must be traversed to reach the network. ( This object is probably best represented as SEQUENCE OF INTEGER. For SMI compatibility, though, it is represented as OCTET STRING. Each AS is represented as a pair of octets according to the following algorithm:

first-byte-of-pair = ASNumber / 256;

second-byte-of-pair = ASNumber & 255;"

::= { bgpPathAttrEntry 4 }

bgpPathAttrNextHop OBJECT-TYPE

SYNTAX IpAddress

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The address of the border router that should be used for the destination network."

::= { bgpPathAttrEntry 5 }

bgpPathAttrInterASMetric OBJECT-TYPE

SYNTAX IpAddress

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The optional inter-AS metric. If this attribute has not been provided for this route, the value for this object is 0."

::= { bgpPathAttrEntry 6 }

bgpEstablished TRAP-TYPE

ENTERPRISE { bgp }

VARIABLES { bgpPeerRemoteAddr,  
bgpPeerLastError,  
bgpPeerState }

DESCRIPTION

"The BGP Established event is generated when the BGP FSM enters the ESTABLISHED state. "

::= 1

```
bgpBackwardTransition TRAP-TYPE
    ENTERPRISE { bgp }
    VARIABLES { bgpPeerRemoteAddr,
                bgpPeerLastError,
                bgpPeerState }
    DESCRIPTION
        "The BGPBackwardTransition Event is generated
        when the BGP FSM moves from a higher numbered
        state to a lower numbered state."
    ::= 2
END
```

## 6. Acknowledgements

We would like to acknowledge the assistance of all the members of the Interconnectivity Working Group, and particularly the following individuals:

Yakov Rekhter, IBM  
Rob Coltun, University of Maryland  
Guy Almes, Rice University  
Jeff Honig, Cornell Theory Center  
Marshall T. Rose, PSI, Inc.  
Dennis Ferguson, University of Toronto  
Mike Mathis, PSC

## 7. References

- [1] Cerf, V., "IAB Recommendations for the Development of Internet Network Management Standards", RFC 1052, NRI, April 1988.
- [2] Cerf, V., "Report of the Second Ad Hoc Network Management Review Group", RFC 1109, NRI, August 1989.
- [3] Rose M., and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based internets", RFC 1155, Performance Systems International, Hughes LAN Systems, May 1990.
- [4] McCloghrie K., and M. Rose, "Management Information Base for Network Management of TCP/IP-based internets", RFC 1156, Hughes LAN Systems, Performance Systems International, May 1990.
- [5] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol", RFC 1157, SNMP Research, Performance Systems International, Performance Systems International, MIT Laboratory for Computer Science, May 1990.
- [6] McCloghrie K., and M. Rose, Editors, "Management Information Base

for Network Management of TCP/IP-based internets", RFC 1213, Performance Systems International, March 1991.

- [7] Information processing systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1), International Organization for Standardization, International Standard 8824, December 1987.
- [8] Information processing systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Notation One (ASN.1), International Organization for Standardization, International Standard 8825, December 1987.
- [9] Rose, M., and K. McCloghrie, Editors, "Concise MIB Definitions", RFC 1212, Performance Systems International, Hughes LAN Systems, March 1991.
- [10] Rose, M., Editor, "A Convention for Defining Traps for use with the SNMP", RFC 1215, Performance Systems International, March 1991.
- [11] Lougheed, K., and Y. Rekhter, "A Border Gateway Protocol 3 (BGP-3)", RFC 1267, cisco Systems, T.J. Watson Research Center, IBM Corp., October 1991.
- [12] Rekhter, Y., and P. Gross, Editors, "Application of the Border Gateway Protocol in the Internet", RFC 1268, T.J. Watson Research Center, IBM Corp., ANS, October 1991.

## 8. Security Considerations

Security issues are not discussed in this memo.

## Authors' Addresses

Steven Willis  
Wellfleet Communications Inc.  
15 Crosby Drive  
Bedford, MA 01730

Phone: (617) 275-2400  
Email: [swillis@wellfleet.com](mailto:swillis@wellfleet.com)

John Burruss  
Wellfleet Communications Inc.  
15 Crosby Drive  
Bedford, MA 01730

Phone: (617) 275-2400  
Email: [jburruss@wellfleet.com](mailto:jburruss@wellfleet.com)