The Definitions of Managed Objects for the Bridge Network Control Protocol of the Point-to-Point Protocol

Status of this Memo

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.

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 describes managed objects used for managing the bridge Network Control Protocol [10] on subnetwork interfaces using the family of Point-to-Point Protocols.

Table of Contents

1. The Network Management Framework ...................... 1
2. Objects ............................................... 2
2.1 Format of Definitions ................................ 2
3. Overview ........................................... 2
3.1 Object Selection Criteria ............................ 2
3.2 Structure of the PPP ................................. 3
3.3 MIB Groups ........................................ 3
4. Definitions .......................................... 4
5. Acknowledgements ..................................... 13
6. Security Considerations ................................ 14
7. References .......................................... 14
8. Author’s Address ..................................... 15

1. The Network Management Framework

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

STD 16/RFC 1155 which defines the SMI, the mechanisms used for describing and naming objects for the purpose of management. STD 16/RFC 1212 defines a more concise description mechanism, which is
wholly consistent with the SMI.

STD 17/RFC 1213 which defines MIB-II, the core set of managed objects for the Internet suite of protocols.

STD 15/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.

2. 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) [3] defined in the SMI. In particular, each object type is named by an OBJECT IDENTIFIER, an administratively assigned name. 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 descriptor, to refer to the object type.

2.1. Format of Definitions

Section 4 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 [5,6].

3. Overview

3.1. Object Selection Criteria

To be consistent with IAB directives and good engineering practice, an explicit attempt was made to keep this MIB as simple as possible. This was accomplished by applying the following criteria to objects proposed for inclusion:

(1) Require objects be essential for either fault or configuration management. In particular, objects for which the sole purpose was to debug implementations were explicitly excluded from the MIB.

(2) Consider evidence of current use and/or utility.

(3) Limit the total number of objects.

(4) Exclude objects which are simply derivable from others in
3.2. Structure of the PPP

This section describes the basic model of PPP used in developing the PPP MIB. This information should be useful to the implementor in understanding some of the basic design decisions of the MIB.

The PPP is not one single protocol but a large family of protocols. Each of these is, in itself, a fairly complex protocol. The PPP protocols may be divided into three rough categories:

Control Protocols
   The Control Protocols are used to control the operation of the PPP. The Control Protocols include the Link Control Protocol (LCP), the Password Authentication Protocol (PAP), the Link Quality Report (LQR), and the Challenge Handshake Authentication Protocol (CHAP).

Network Protocols
   The Network Protocols are used to move the network traffic over the PPP interface. A Network Protocol encapsulates the datagrams of a specific higher-layer protocol that is using the PPP as a data link. Note that within the context of PPP, the term "Network Protocol" does not imply an OSI Layer-3 protocol; for instance, there is a Bridging network protocol.

Network Control Protocols (NCPs)
   The NCPs are used to control the operation of the Network Protocols. Generally, each Network Protocol has its own Network Control Protocol; thus, the IP Network Protocol has its IP Control Protocol, the Bridging Network Protocol has its Bridging Network Control Protocol and so on.

This document specifies the objects used in managing one of these protocols, namely the Bridge Network Control Protocol.

3.3. MIB Groups

Objects in this MIB are arranged into several MIB groups. Each group is organized as a set of related objects.

These groups are the basic unit of conformance: if the semantics of a group are applicable to an implementation then all objects in the group must be implemented.

The PPP MIB is organized into several MIB Groups, including, but not limited to, the following groups:
This document specifies the following group:

The PPP Bridge Group

The PPP Bridge Group contains configuration, status, and control variables that apply to the operation of Bridging over PPP.

Implementation of this group is mandatory for all implementations of PPP that support the Bridging over PPP.

4. Definitions

PPP-BRIDGE-NCP-MIB DEFINITIONS ::= BEGIN

IMPORTS
  Counter
  FROM RFC1155-SMI
  ifIndex
  FROM RFC1213-MIB
  OBJECT-TYPE
      FROM RFC-1212
  ppp
      FROM PPP-LCP-MIB;

pppBridge OBJECT IDENTIFIER ::= { ppp 4 }

-- The PPP Bridge NCP Group.
-- Implementation of this group is mandatory for all
-- PPP implementations that support MAC Bridging over
-- PPP (RFC1220).
--

-- The following object reflect the values of the option
-- parameters used in the PPP Link Control Protocol
-- pppBridgeLocalToRemoteTinygramCompression
-- pppBridgeRemoteToLocalTinygramCompression
-- pppBridgeLocalToRemoteLanId
-- pppBridgeRemoteToLocalLanId
-- These values are not available until after the PPP Option
-- negotiation has completed, which is indicated by the link
-- reaching the open state (i.e. pppBridgeOperStatus is set to
-- opened).
--
-- Therefore, when pppBridgeOperStatus is not opened
-- the contents of these objects is undefined. The value
-- returned when accessing the objects is an implementation
-- dependent issue.

pppBridgeTable   OBJECT-TYPE
SYNTAX    SEQUENCE OF PppBridgeEntry
ACCESS    not-accessible
STATUS    mandatory
DESCRIPTION
"Table containing the parameters and statistics
for the local PPP entity that are related to
the operation of Bridging over the PPP."
 ::= { pppBridge 1 }

pppBridgeEntry   OBJECT-TYPE
SYNTAX    PppBridgeEntry
ACCESS    not-accessible
STATUS    mandatory
DESCRIPTION
"Bridging information for a particular PPP
link."
INDEX     { ifIndex }
 ::= { pppBridgeTable 1 }

PppBridgeEntry ::= SEQUENCE {  
  pppBridgeOperStatus  
  INTEGER,  
  pppBridgeLocalToRemoteTinygramCompression  
  INTEGER,  
  pppBridgeRemoteToLocalTinygramCompression  
  INTEGER,  
  pppBridgeLocalToRemoteLanId  
  INTEGER,  
  pppBridgeRemoteToLocalLanId  
  INTEGER  
}

pppBridgeOperStatus   OBJECT-TYPE
SYNTAX    INTEGER {opened(1), not-opened(2)}
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
"The operational status of the Bridge network protocol. If the value of this object is up then the finite state machine for the Bridge network protocol has reached the Opened state."
::= { pppBridgeEntry 1 }

pppBridgeLocalToRemoteTinygramCompression   OBJECT-TYPE
SYNTAX    INTEGER { false(1), true(2) }
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
"Indicates whether the local node will perform Tinygram Compression when sending packets to the remote entity. If false then the local entity will not perform Tinygram Compression. If true then the local entity will perform Tinygram Compression. The value of this object is meaningful only when the link has reached the open state (pppBridgeOperStatus is opened)."
REFERENCE
"Section 6.7, Tinygram Compression Option, of RFC1220"
::= { pppBridgeEntry 2 }

pppBridgeRemoteToLocalTinygramCompression   OBJECT-TYPE
SYNTAX    INTEGER { false(1), true(2) }
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
"If false(1) then the remote entity is not expected to perform Tinygram Compression. If true then the remote entity is expected to perform Tinygram Compression. The value of this object is meaningful only when the link has reached the open state (pppBridgeOperStatus is opened)."
REFERENCE
"Section 6.7, Tinygram Compression Option, of RFC1220"
::= { pppBridgeEntry 3 }
**pppBridgeLocalToRemoteLanId**  OBJECT-TYPE
SYNTAX    INTEGER { false(1), true(2) }
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
"Indicates whether the local node will include
the LAN Identification field in transmitted
packets or not. If false(1) then the local node
will not transmit this field, true(2) means
that the field will be transmitted. The value
of this object is meaningful only when the link
has reached the open state (pppBridgeOperStatus
is opened)."

REFERENCE
"Section 6.8, LAN Identification Option, of
RFC1220"
::= { pppBridgeEntry 4 }

**pppBridgeRemoteToLocalLanId**  OBJECT-TYPE
SYNTAX    INTEGER { false(1), true(2) }
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
"Indicates whether the remote node has
indicated that it will include the LAN
Identification field in transmitted packets or
not. If false(1) then the field will not be
transmitted, if true(2) then the field will be
transmitted. The value of this object is
meaningful only when the link has reached the
open state (pppBridgeOperStatus is opened)."

REFERENCE
"Section 6.8, LAN Identification Option, of
RFC1220"
::= { pppBridgeEntry 5 }

--
-- The PPP Bridge Configuration table
--

**pppBridgeConfigTable**  OBJECT-TYPE
SYNTAX    SEQUENCE OF PppBridgeConfigEntry
ACCESS    not-accessible
STATUS    mandatory
DESCRIPTION
"Table containing the parameters and statistics
...
for the local PPP entity that are related to
the operation of Bridging over the PPP.

::= { pppBridge 2 }

pppBridgeConfigEntry OBJECT-TYPE
SYNTAX    PppBridgeConfigEntry
ACCESS    not-accessible
STATUS    mandatory
DESCRIPTION
"Bridging Configuration information for a
particular PPP link."
INDEX     { ifIndex }
::= { pppBridgeConfigTable 1 }

PppBridgeConfigEntry ::= SEQUENCE {
    pppBridgeConfigAdminStatus
        INTEGER,
    pppBridgeConfigTinygram
        INTEGER,
    pppBridgeConfigRingId
        INTEGER,
    pppBridgeConfigLineId
        INTEGER,
    pppBridgeConfigLanId
        INTEGER
}

pppBridgeConfigAdminStatus OBJECT-TYPE
SYNTAX    INTEGER { open(1), close(2) }
ACCESS    read-write
STATUS    mandatory
DESCRIPTION
"The immediate desired status of the Bridging
network protocol. Setting this object to open
will inject an administrative open event into
the Bridging network protocol’s finite state
machine. Setting this object to close will
inject an administrative close event into the
Bridging network protocol’s finite state
machine."
::= { pppBridgeConfigEntry 1 }

pppBridgeConfigTinygram OBJECT-TYPE
SYNTAX    INTEGER { false(1), true(2) }
ACCESS    read-write
STATUS    mandatory
DESCRIPTION
"If false then the local BNCP entity will not
initiate the Tinygram Compression Option
Negotiation. If true then the local BNCP entity
will initiate negotiation of this option."
REFERENCE
"Section 6.7, Tinygram Compression Option, of
RFC1220"
DEFVAL    { true }
::= { pppBridgeConfigEntry 2 }

pppBridgeConfigRingId   OBJECT-TYPE
SYNTAX    INTEGER { false(1), true(2) }
ACCESS    read-write
STATUS    mandatory
DESCRIPTION
"If false then the local PPP Entity will not
initiate a Remote Ring Identification Option
negotiation. If true then the local PPP entity
will initiate this negotiation. This MIB object
is relevant only if the interface is for 802.5
Token Ring bridging."
REFERENCE
"Section 6.4, IEEE 802.5 Remote Ring
Identification Option, of RFC1220"
DEFVAL    { false }
::= { pppBridgeConfigEntry 3 }

pppBridgeConfigLineId   OBJECT-TYPE
SYNTAX    INTEGER { false(1), true(2) }
ACCESS    read-write
STATUS    mandatory
DESCRIPTION
"If false then the local PPP Entity is not to
initiate a Line Identification Option
negotiation. If true then the local PPP entity
will initiate this negotiation. This MIB object
is relevant only if the interface is for 802.5
Token Ring bridging."
REFERENCE
"Section 6.5, IEEE 802.5 Line Identification
Option, of RFC1220"
DEFVAL    { false }
::= { pppBridgeConfigEntry 4 }
pppBridgeConfigLanId OBJECT-TYPE
SYNTAX INTEGER { false(1), true(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION "If false then the local BNCP entity will not
initiate the LAN Identification Option
Negotiation. If true then the local BNCP entity
will initiate negotiation of this option."
REFERENCE
"Section 6.8, LAN Identification Option, of
RFC1220"
DEFVAL { false }
 ::= { pppBridgeConfigEntry 5 }

-- The PPP Bridge Media Status Table --

pppBridgeMediaTable OBJECT-TYPE
SYNTAX SEQUENCE OF PppBridgeMediaEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION "Table identifying which MAC media types are
enabled for the Bridging NCPs."
 ::= { pppBridge 3 }

PppBridgeMediaEntry OBJECT-TYPE
SYNTAX PppBridgeMediaEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION "Status of a specific MAC Type for a specific
PPP Link."
INDEX { ifIndex, pppBridgeMediaMacType }
 ::= { pppBridgeMediaTable 1 }

PppBridgeMediaEntry ::= SEQUENCE {
  pppBridgeMediaMacType
    INTEGER,
  pppBridgeMediaLocalStatus
    INTEGER,
  pppBridgeMediaRemoteStatus
    INTEGER
}
pppBridgeMediaMacType OBJECT-TYPE
SYNTAX INTEGER(0..2147483647)
ACCESS read-only
STATUS mandatory
DESCRIPTION
"The MAC type for which this entry in the pppBridgeMediaTable is providing status information. Valid values for this object are defined in Section 6.6 MAC Type Support Selection of RFC1220 (Bridging Point-to-Point Protocol)."
REFERENCE
"Section 6.6, MAC Type Support Selection, of RFC1212."
::= { pppBridgeMediaEntry 1 }

pppBridgeMediaLocalStatus OBJECT-TYPE
SYNTAX INTEGER { accept(1), dont-accept(2) }
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Indicates whether the local PPP Bridging Entity will accept packets of the protocol type identified in pppBridgeMediaMacType on the PPP link identified by ifIndex or not. If this object is accept then any packets of the indicated MAC type will be received and properly processed. If this object is dont-accept then received packets of the indicated MAC type will not be properly processed."
REFERENCE
"Section 6.6, MAC Type Support Selection, of RFC1212."
::= { pppBridgeMediaEntry 2 }

pppBridgeMediaRemoteStatus OBJECT-TYPE
SYNTAX INTEGER { accept(1), dont-accept(2) }
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Indicates whether the local PPP Bridging Entity believes that the remote PPP Bridging Entity will accept packets of the protocol type identified in pppBridgeMediaMacType on the PPP
link identified by ifIndex or not."
REFERENCE  "Section 6.6, MAC Type Support Selection, of RFC1212."
::= { pppBridgeMediaEntry 3 }

--
-- The PPP Bridge Media Configuration Table
--

pppBridgeMediaConfigTable   OBJECT-TYPE
SYNTAX    SEQUENCE OF PppBridgeMediaConfigEntry
ACCESS    not-accessible
STATUS    mandatory
DESCRIPTION
"Table identifying which MAC media types are enabled for the Bridging NCPs."
::= { pppBridge 4 }

pppBridgeMediaConfigEntry   OBJECT-TYPE
SYNTAX    PppBridgeMediaConfigEntry
ACCESS    not-accessible
STATUS    mandatory
DESCRIPTION
"Status of a specific MAC Type for a specific PPP Link."
INDEX     { ifIndex, pppBridgeMediaConfigMacType }
::= { pppBridgeMediaConfigTable 1 }

PppBridgeMediaConfigEntry ::= SEQUENCE {
  pppBridgeMediaConfigMacType
    INTEGER,
  pppBridgeMediaConfigLocalStatus
    INTEGER
}

pppBridgeMediaConfigMacType   OBJECT-TYPE
SYNTAX    INTEGER(0..2147483647)
ACCESS    read-write
STATUS    mandatory
DESCRIPTION
"The MAC type for which this entry in the pppBridgeMediaConfigTable is providing status information. Valid values for this object are
defined in Section 6.6 MAC Type Support Selection of RFC1220 (Bridging Point-to-Point Protocol)."

REFERENCE
"Section 6.6, MAC Type Support Selection, of RFC1212."
::= { pppBridgeMediaConfigEntry 1 }

pppBridgeMediaConfigLocalStatus OBJECT-TYPE
SYNTAX INTEGER { accept(1), dont-accept(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION "Indicates whether the local PPP Bridging Entity should accept packets of the protocol type identified in pppBridgeMediaConfigMacType on the PPP link identified by ifIndex or not. Setting this object to the value dont-accept has the affect of invalidating the corresponding entry in the pppBridgeMediaConfigTable object. 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. Changing this object will have effect when the link is next restarted."

REFERENCE
"Section 6.6, MAC Type Support Selection, of RFC1212."
::= { pppBridgeMediaConfigEntry 2 }

END

6. Acknowledgements

This document was produced by the PPP working group. In addition to the working group, the author wishes to thank the following individuals for their comments and contributions:

Bill Simpson -- Daydreamer
Glenn McGregor -- Merit
Jesse Walker -- DEC
Chris Gunner -- DEC
6. Security Considerations

The PPP MIB affords the network operator the ability to configure and control the PPP links of a particular system, including the PPP authentication protocols. This represents a security risk.

These risks are addressed in the following manners:

(1) All variables which represent a significant security risk are placed in separate, optional, MIB Groups. As the MIB Group is the quantum of implementation within a MIB, the implementor of the MIB may elect not to implement these groups.

(2) The implementor may choose to implement the variables which present a security risk so that they may not be written, i.e., the variables are READ-ONLY. This method still presents a security risk, and is not recommended, in that the variables, specifically the PPP Authentication Protocols’ variables, may be easily read.

(3) Using SNMPv2, the operator can place the variables into MIB views which are protected in that the parties which have access to those MIB views use authentication and privacy protocols, or the operator may elect to make these views not accessible to any party. In order to facilitate this placement, all security-related variables are placed in separate MIB Tables. This eases the identification of the necessary MIB View Subtree.

7. References


8. Author’s Address

Frank Kastenholz
FTP Software, Inc.
2 High Street
North Andover, Mass 01845 USA

Phone: (508) 685-4000
EMail: kasten@ftp.com