The Definitions of Managed Objects for
the Security Protocols 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 Security Protocols on subnetwork interfaces using the family of Point-to-Point Protocols [8, 9, 10, 11, & 12].

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 ............................... 2
3.3 MIB Groups ....................................... 3
4. Definitions .......................................... 4
5. Acknowledgements .................................. 9
6. Security Considerations ............................ 10
7. References ....................................... 11
8. Author’s Address .................................. 12

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
this or other MIBs.

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 PPP Authentication Protocols.

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:

PPP Security Group

The PPP Security Group contains all configuration and control variables that apply to PPP security.

Implementation of this group is optional. Implementation is optional since the variables in this group provide configuration and control for the PPP Security functions. Thus, these variables should be protected by SNMPv2 security. If an agent does not support SNMPv2 with privacy it is strongly advised that this group not be implemented. See the section on "Security Considerations" at the end of this document.

4. Definitions

PPP-SEC-MIB DEFINITIONS ::= BEGIN

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

pppSecurity OBJECT IDENTIFIER ::= { ppp 2 }

pppSecurityProtocols OBJECT IDENTIFIER ::= { pppSecurity 1 }

-- The following uniquely identify the various protocols
-- used by PPP security. These OBJECT IDENTIFIERS are
-- used in the pppSecurityConfigProtocol and
-- pppSecuritySecretsProtocol objects to identify to which
-- protocols the table entries apply.

pppSecurityPapProtocol OBJECT IDENTIFIER ::= { pppSecurityProtocols 1 }
pppSecurityChapMD5Protocol OBJECT IDENTIFIER ::= { pppSecurityProtocols 2 }
-- PPP Security Group
-- Implementation of this group is optional.

-- This table allows the network manager to configure
-- which security protocols are to be used on which
-- link and in what order of preference each is to be tried

pppSecurityConfigTable OBJECT-TYPE
SYNTAX SEQUENCE OF PppSecurityConfigEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION
"Table containing the configuration and
preference parameters for PPP Security."
::= { pppSecurity 2 }

PppSecurityConfigEntry ::= SEQUENCE {
  pppSecurityConfigLink
  INTEGER,
  pppSecurityConfigPreference
  INTEGER,
  pppSecurityConfigProtocol
  OBJECT IDENTIFIER,
  pppSecurityConfigStatus
  INTEGER
}

pppSecurityConfigLink OBJECT-TYPE
SYNTAX INTEGER(0..2147483647)
ACCESS read-write
STATUS mandatory
DESCRIPTION
"The value of ifIndex that identifies the entry
in the interface table that is associated with
the local PPP entity’s link for which this
particular security algorithm shall be
attempted. A value of 0 indicates the default
algorithm - i.e., this entry applies to all
links for which explicit entries in the table
do not exist."
::= { pppSecurityConfigEntry 1 }

pppSecurityConfigPreference  OBJECT-TYPE
SYNTAX    INTEGER(0..2147483647)
ACCESS    read-write
STATUS    mandatory
DESCRIPTION
"The relative preference of the security
protocol identified by
with lower values of
pppSecurityConfigPreference are tried before
protocols with higher values of
pppSecurityConfigPreference."
::= { pppSecurityConfigEntry 2 }

pppSecurityConfigProtocol  OBJECT-TYPE
SYNTAX    OBJECT IDENTIFIER
ACCESS    read-write
STATUS    mandatory
DESCRIPTION
"Identifies the security protocol to be
attempted on the link identified by
pppSecurityConfigLink at the preference level
identified by pppSecurityConfigPreference."
::= { pppSecurityConfigEntry 3 }

pppSecurityConfigStatus  OBJECT-TYPE
SYNTAX    INTEGER  {
  invalid(1),
  valid(2)
}
ACCESS    read-write
STATUS    mandatory
DESCRIPTION
"Setting this object to the value invalid(1)
has the effect of invalidating the
corresponding entry in the
pppSecurityConfigTable. 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 pppSecurityConfigStatus object.

DEFVAL { valid }
::= { pppSecurityConfigEntry 4 }

-- This table contains all of the ID/Secret pair information.

pppSecuritySecretsTable  OBJECT-TYPE
SYNTAX    SEQUENCE OF PppSecuritySecretsEntry
ACCESS    not-accessible
STATUS    mandatory
DESCRIPTION
"Table containing the identities and secrets used by the PPP authentication protocols. As this table contains secret information, it is expected that access to this table be limited to those SNMP Party-Pairs for which a privacy protocol is in use for all SNMP messages that the parties exchange. This table contains both the ID and secret pair(s) that the local PPP entity will advertise to the remote entity and the pair(s) that the local entity will expect from the remote entity. This table allows for multiple id/secret password pairs to be specified for a particular link by using the pppSecuritySecretsIdIndex object."
::= { pppSecurity 3 }

pppSecuritySecretsEntry  OBJECT-TYPE
SYNTAX    PppSecuritySecretsEntry
ACCESS    not-accessible
STATUS    mandatory
DESCRIPTION
"Secret information."
INDEX { pppSecuritySecretsLink,
         pppSecuritySecretsIdIndex }
::= { pppSecuritySecretsTable 1 }
PppSecuritySecretsEntry ::= SEQUENCE {
    pppSecuritySecretsLink
        INTEGER,
    pppSecuritySecretsIdIndex
        INTEGER,
    pppSecuritySecretsDirection
        INTEGER,
    pppSecuritySecretsProtocol
        OBJECT IDENTIFIER,
    pppSecuritySecretsIdentity
        OCTET STRING,
    pppSecuritySecretsSecret
        OCTET STRING,
    pppSecuritySecretsStatus
        INTEGER
}

pppSecuritySecretsLink   OBJECT-TYPE
SYNTAX    INTEGER(0..2147483647)
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "The link to which this ID/Secret pair applies.  By
     convention, if the value of this object is 0 then
     the ID/Secret pair applies to all links."
::= { pppSecuritySecretsEntry 1 }

pppSecuritySecretsIdIndex   OBJECT-TYPE
SYNTAX    INTEGER(0..2147483647)
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "A unique value for each ID/Secret pair that has
     been defined for use on this link.  This allows
     multiple ID/Secret pairs to be defined for each
     link.  How the local entity selects which pair
     to use is a local implementation decision."
::= { pppSecuritySecretsEntry 2 }

pppSecuritySecretsDirection   OBJECT-TYPE
SYNTAX    INTEGER { local-to-remote(1),
                        remote-to-local(2) }
ACCESS    read-write
STATUS    mandatory
DESCRIPTION
"This object defines the direction in which a particular ID/Secret pair is valid. If this object is local-to-remote then the local PPP entity will use the ID/Secret pair when attempting to authenticate the local PPP entity to the remote PPP entity. If this object is remote-to-local then the local PPP entity will expect the ID/Secret pair to be used by the remote PPP entity when the remote PPP entity attempts to authenticate itself to the local PPP entity."

::= { pppSecuritySecretsEntry 3 }

pppSecuritySecretsProtocol OBJECT-TYPE
SYNTAX OBJECT IDENTIFIER
ACCESS read-write
STATUS    mandatory
DESCRIPTION
"The security protocol (e.g. CHAP or PAP) to which this ID/Secret pair applies."

::= { pppSecuritySecretsEntry 4 }

pppSecuritySecretsIdentity OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(0..255))
ACCESS read-write
STATUS    mandatory
DESCRIPTION
"The Identity of the ID/Secret pair. The actual format, semantics, and use of pppSecuritySecretsIdentity depends on the actual security protocol used. For example, if pppSecuritySecretsProtocol is pppSecurityPapProtocol then this object will contain a PAP Peer-ID. If pppSecuritySecretsProtocol is pppSecurityChapMD5Protocol then this object would contain the CHAP NAME parameter."

::= { pppSecuritySecretsEntry 5 }

pppSecuritySecretsSecret OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(0..255))
ACCESS read-write
STATUS    mandatory
DESCRIPTION
"The secret of the ID/Secret pair. The actual format, semantics, and use of pppSecuritySecretsSecret depends on the actual security protocol used. For example, if pppSecuritySecretsProtocol is pppSecurityPapProtocol then this object will contain a PAP Password. If pppSecuritySecretsProtocol is pppSecurityChapMD5Protocol then this object would contain the CHAP MD5 Secret."

::= { pppSecuritySecretsEntry 6 }

pppSecuritySecretsStatus OBJECT-TYPE
SYNTAX INTEGER {
  invalid(1),
  valid(2)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Setting this object to the value invalid(1) has the effect of invalidating the corresponding entry in the pppSecuritySecretsTable. 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 pppSecuritySecretsStatus object."

DEFVAL { valid }
::= { pppSecuritySecretsEntry 7 }

END

5. 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
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.

4. The PPP Security Protocols MIB (this document) contains several objects which are very sensitive from a security point of view.

Specifically, this MIB contains objects that define the PPP Peer Identities (which can be viewed as "userids") and the secrets used to authenticate those Peer Identities (similar to a "password" for the "userid").

Also, this MIB contains variables which would allow a network manager to control the operation of the security features of PPP. An intruder could disable PPP security if these variables were not properly protected.

Thus, in order to preserve the integrity, security and privacy of the
PPP security features, an implementation will allow access to this MIB only via SNMPv2 and then only for parties which are privacy enhanced. Other access modes, e.g., SNMPv1 or SNMPv2 without privacy enhancement, are very dangerous and the security of the PPP service may be compromised.

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