Network Working Group   M. Krishnaswamy
Request for Comments: 3055   Photuris, Inc.
Category: Standards Track   D. Romascanu
                                                      Avaya Communication
                                                            February 2001

        Management Information Base for the PINT Services Architecture

Status of this Memo

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

Abstract

   This memo describes a proposed Management Information Base (MIB) for
   the PSTN/Internet Interworking (PINT) Services Architecture.

Table of Contents

1. Introduction

   PINT services are an emerging set of new Internet based applications
   where voice (and fax) requests to the PSTN (Public Switched Telephone
   Network) are carried over the Internet. RFC 2458 [1] gives a good
   introduction to the (pre-standard) PINT architecture and services.
   It also has examples of some of the early implementations of pre-
   PINT.

   This document defines a MIB which contains the elements for
   monitoring the performance of a PINT based service. The MIB consists
   of details of the four basic PINT services and their performance
   statistics measured under various criteria.

   It is not the purpose of this MIB to enable management of the PINT
   networking elements. We are concerned only with the PINT specific
   performance parameters. While it is understood that PINT service
   performance is closely related to host and network performance, they
   are not addressed here.

2. The SNMP Management Framework

   The SNMP Management Framework presently consists of five major
   components:

   o   An overall architecture, described in RFC 2571 [2].

   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 [3], STD 16, RFC 1212 [4] and RFC 1215 [5].
        The second version, called SMIv2, is described in STD 58,
        RFC 2578 [6], RFC 2579 [7] and RFC 2580 [8].

   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 [9]. A second version of the SNMP
        message protocol, which is not an Internet standards track
        protocol, is called SNMPv2c and described in RFC 1901 [10] and
        RFC 1906 [11]. The third version of the message protocol is
        called SNMPv3 and described in RFC 1906 [11], RFC 2572 [12] and
        RFC 2574 [13].

   o   Protocol operations for accessing management information. The
        first set of protocol operations and associated PDU formats is
        described in STD 15, RFC 1157 [9]. A second set of protocol
        operations and associated PDU formats is described in RFC 1905
        [14].

   o   A set of fundamental applications described in RFC 2573 [15] and
        the view-based access control mechanism described in RFC 2575
        [16].

   A more detailed introduction to the current SNMP Management Framework
   can be found in RFC 2570 [17].

   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.

3. The need for PINT services monitoring MIB

   Traditionally voice (and fax) requests originate and terminate inside
   a PSTN network. This network is well known for robust handling of
   the requests, in terms of availability and security. However when
   the requests originate from the Internet there is a concern both on
   the part of the user as well as the provider about issues like
   reliable forwarding of the call requests to the PINT gateway under
   various network conditions, user/host authentication, secure handling
   of the user information etc. Performance and security management
   becomes all the more important where PINT services cross multiple
   administrative domains (or providers).

   This MIB is an attempt to list the parameters that need to be
   monitored on an user, PINT client, PINT server and PINT gateway
   basis.

   (PINT services, their invocation methods/protocols and security
   issues associated with the PINT architecture are discussed in detail
   in [18]).

4. PINT MIB - Overview

   Following is a list of some explanations on the MIB definitions that
   we have chosen to construct.

   o   The basic purpose of this MIB is to monitor the access to PINT
       services both from the performance and security point of view.
       Information may pertain to a certain user or his/her system
       (PINT client) or the system providing the PINT services (PINT
       server) or the PINT gateway that forwards the call to the PSTN
       network.

   o   We chose to build the configuration table as an extension of the
       Application MIB - RFC 2287 [19] using the augments construct.
       Server location and contact might be retrieved from the standard
       MIB-II sysLocation and sysContact objects. There is no need to
       replicate this information in the PINT MIB. However, the PINT
       administrator may be a different person than the sysadmin with
       global responsibilities, thus a pintSysContact object is
       defined.

   o   We chose to monitor the gateway connections from the PINT
       server. While the agent runs in the PINT servers, the
       connections to the gateways might need to be monitored in order
       to understand what goes on. We placed them in a separate MIB
       group, and by using MODULE-COMPLIANCE clauses, agents that
       cannot implement this stuff will not be mandated to do it.

   o   There is no traps definition in this MIB module. Note that
       thresholding on counters is always possible by using a standard
       mechanism defined by the Remote Monitoring MIB, that can be
       referenced here. Some events that may be defined by using this
       mechanisms:

           *   continuous login/authentication failure or refusal from a
               particular client or user

           *   nuisance call - repeated calls (within a specified
               period) to a number originating from the same user

   o   The client performance and user performance tables may be rather
       resource demanding for an agent implementation. In some MIBs,
       like the Remote Monitoring (RMON) MIBs, control mechanisms were
       built in order to activate those statistics on demand. If
       needed, a sorting ('topN') mechanism can be designed, so that a
       sorted view of clients or users is presented for the high level
       debugging.

    o   We built a time-distribution trying to cover both short-lived,
        as well as longer sessions (1-10 secs, 10 secs - 1 min., 1-15
        min., 15 mins-24 hours, longer).

    o   PintServerClientAddress is defined as a SnmpAdminString. It may
        include an IpAddress and/or name, but we preferred to minimize
        the number of indices at this stage, and keep a human-readable
        format at the same time.

    o   We define pintServerUserIdName as the UserId. This UserId needs
        to be unique across multiple PINT servers and gateways
        (depending on the architecture) and is mapped to the SessionId.
        One way to achieve this uniqueness is by appending clientId to
        the UserId string before sending to the PINT server. The
        SessionId could then be a combination of this new UserId and a
        timestamp.

5. Definitions

PINT-MIB DEFINITIONS ::= BEGIN

        IMPORTS
         OBJECT-TYPE, Counter32, MODULE-IDENTITY, mib-2
        FROM   SNMPv2-SMI
        TEXTUAL-CONVENTION
        FROM SNMPv2-TC
        MODULE-COMPLIANCE, OBJECT-GROUP
        FROM SNMPv2-CONF
        sysApplInstallPkgEntry
        FROM SYSAPPL-MIB
        SnmpAdminString
        FROM SNMP-FRAMEWORK-MIB;   -- RFC 2571 [2]

        pintMib MODULE-IDENTITY
        LAST-UPDATED "200102010000Z" -- 1 Feb 2001

ORGANIZATION "IETF PINT Working Group"
CONTACT-INFO "
        Chairs: Steve Bellovin
                    E-mail: smb@research.att.com

                    Igor Faynberg
                    E-mail: faynberg@lucent.com

        Authors: Murali Krishnaswamy
                    Postal: 20 Corporate Place South
                            Piscataway, NJ 08854
                            Tel: +1 (732)465-1000

                             E-mail: murali@photuris.com

                             Dan Romascanu
                             Postal: Atidim Technology Park, Bldg 3
                             Tel Aviv, Israel
                             Tel: +972 3 6458414
                             E-mail: dromasca@avaya.com

         General Discussion:pint@lists.bell-labs.com
         To Subscribe: pint-request@lists.bell-labs.com
         In Body: subscribe your-email-addres
         Archive: http://www.bell-labs.com/mailing-lists/pint/
         "

DESCRIPTION
    "This MIB defines the objects necessary to monitor
     PINT Services"

-- Revision history

REVISION "200102010000Z" -- 1 Feb 2001
DESCRIPTION
    "Initial version, published as RFC 3055."
::= { mib-2 93 }

PintServiceType ::= TEXTUAL-CONVENTION
        STATUS   current
        DESCRIPTION
            "This TC describes the type of a PINT service."
        SYNTAX INTEGER {
                r2C(1),   -- Request-to-Talk
                r2F(2),   -- Request-to-Fax
                r2FB(3),   -- Request-to-Fax-Back
                r2HC(4)   -- Request-to-Hear-Content
        }

PintPerfStatPeriod ::= TEXTUAL-CONVENTION
STATUS   current
DESCRIPTION
    "This TC describes the statistics period of time.

    Note that the values of the counters indexed with a value
    SinceReboot(4) can be potentially affected by a counter rollover.
    It is the responsibility of the application using this object to
    take into account that the counter has been zeroed each time it
    reached a value of (2**32-1)."
SYNTAX INTEGER {
last30sec(1),   -- Performance Statics for the last 30 sec

```
last15min(2),    --     15 min
last24Hr(3),     --     24 Hour
sinceReboot(4)   --     Since the time the pint server was
--      last rebooted
}

pintServerConfig        OBJECT IDENTIFIER ::= { pintMib 1 }
pintServerMonitor       OBJECT IDENTIFIER ::= { pintMib 2 }
pintMibConformance      OBJECT IDENTIFIER ::= { pintMib 3 }

-- pintServerConfig - PINT configuration MIB variables

pintReleaseNumber OBJECT-TYPE
    SYNTAX       SnmpAdminString
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
     "An indication of version of the PINT protocol supported
      by this agent."
    ::= { pintServerConfig 1 }

pintSysContact          OBJECT-TYPE
    SYNTAX       SnmpAdminString
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
     "Contact information related to the administration of the PINT
      services."
    ::= { pintServerConfig 2 }

pintApplInstallPkgTable OBJECT-TYPE
    SYNTAX       SEQUENCE OF PintApplInstallPkgEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
     "Table describing the PINT applications that are installed."
    ::= { pintServerConfig 3 }

pintApplInstallPkgEntry OBJECT-TYPE
    SYNTAX       PintApplInstallPkgEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
     "Entries per PINT Application."
    AUGMENTS { sysApplInstallPkgEntry }
    ::= { pintApplInstallPkgTable 1 }

PintApplInstallPkgEntry ::= SEQUENCE {
```

```
pintApplInstallPkgDescription    SnmpAdminString
}

pintApplInstallPkgDescription OBJECT-TYPE
    SYNTAX        SnmpAdminString
    MAX-ACCESS  read-only
    STATUS        current
    DESCRIPTION
     "Textual description of the installed PINT application."
    ::= { pintApplInstallPkgEntry 1 }

pintRegisteredGatewayTable OBJECT-TYPE
    SYNTAX        SEQUENCE OF PintRegisteredGatewayEntry
    MAX-ACCESS  not-accessible
    STATUS        current
    DESCRIPTION
     "Table describing the registered gateway applications."
    ::= { pintServerConfig 4 }

pintRegisteredGatewayEntry OBJECT-TYPE
    SYNTAX        PintRegisteredGatewayEntry
    MAX-ACCESS  not-accessible
    STATUS        current
    DESCRIPTION
     "Entries per Registered Gateway Application."
AUGMENTS { sysApplInstallPkgEntry  }
    ::= { pintRegisteredGatewayTable 1 }

PintRegisteredGatewayEntry ::= SEQUENCE {
pintRegisteredGatewayName        SnmpAdminString,
pintRegisteredGatewayDescription SnmpAdminString
}

pintRegisteredGatewayName OBJECT-TYPE
    SYNTAX        SnmpAdminString
    MAX-ACCESS read-only
    STATUS        current
    DESCRIPTION
     "Name of the registered gateway."
    ::= { pintRegisteredGatewayEntry 1 }

pintRegisteredGatewayDescription OBJECT-TYPE
    SYNTAX        SnmpAdminString
    MAX-ACCESS  read-only
    STATUS        current
    DESCRIPTION
     "Textual description of the registered gateway."
    ::= { pintRegisteredGatewayEntry 2 }
```

-- pintServerMonitor - PINT monitoring statistics MIB variables

pintServerGlobalPerf   OBJECT IDENTIFIER ::= {pintServerMonitor 1 }
pintServerClientPerf   OBJECT IDENTIFIER ::= {pintServerMonitor 2 }
pintServerUserIdPerf   OBJECT IDENTIFIER ::= {pintServerMonitor 3 }
pintServerGatewayPerf   OBJECT IDENTIFIER ::= {pintServerMonitor 4 }

pintServerGlobalStatsTable   OBJECT-TYPE
    SYNTAX   SEQUENCE OF PintServerGlobalStatsEntry
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
     "Table displaying the monitored global server statistics."
    ::= { pintServerGlobalPerf 1 }

pintServerGlobalStatsEntry OBJECT-TYPE
    SYNTAX   PintServerGlobalStatsEntry
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
     "Entries in the global statistics table.
     One entry is defined for each monitored service type and
     performance statistics collection period."
    INDEX {pintServerServiceTypeIndex, pintServerPerfStatPeriodIndex}
    ::= { pintServerGlobalStatsTable 1 }

PintServerGlobalStatsEntry   ::= SEQUENCE {
pintServerServiceTypeIndex   PintServiceType,
pintServerPerfStatPeriodIndex   PintPerfStatPeriod,
pintServerGlobalCallsReceived   Counter32,
pintServerGlobalSuccessfulCalls   Counter32,
pintServerGlobalDisconnectedCalls   Counter32,
pintServerGlobalDisCUAutFCalls   Counter32,
pintServerGlobalDisServProbCalls   Counter32,
pintServerGlobalDisGatProbCalls   Counter32
}

pintServerServiceTypeIndex OBJECT-TYPE
    SYNTAX   PintServiceType
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
     "The unique identifier of the monitored service."
    ::= { pintServerGlobalStatsEntry 1 }

pintServerPerfStatPeriodIndex OBJECT-TYPE
    SYNTAX   PintPerfStatPeriod
    MAX-ACCESS not-accessible

```
    STATUS      current
    DESCRIPTION
     "Time period for which the performance statistics are requested
     from the pint server."
    ::= { pintServerGlobalStatsEntry 2 }


pintServerGlobalCallsReceived OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of received global calls."
    ::= { pintServerGlobalStatsEntry 3 }


pintServerGlobalSuccessfulCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of global successful calls."
    ::= { pintServerGlobalStatsEntry 4 }


pintServerGlobalDisconnectedCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of global disconnected (failed) calls."
    ::= { pintServerGlobalStatsEntry 5 }


pintServerGlobalDisCUAutFCalls
OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS read-only
STATUS      current
DESCRIPTION
    "Number of global calls that were disconnected because of client
    or user authorization failure."
::= { pintServerGlobalStatsEntry 6 }


pintServerGlobalDisServProbCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of global calls that were disconnected because of
     server problems."
    ::= { pintServerGlobalStatsEntry 7 }
```

```
pintServerGlobalDisGatProbCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of global calls that were disconnected because of
     gateway problems."
    ::= { pintServerGlobalStatsEntry 8 }

pintServerClientStatsTable      OBJECT-TYPE
    SYNTAX      SEQUENCE OF PintServerClientStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
     "Table displaying the monitored server client statistics."
    ::= { pintServerClientPerf 1 }

pintServerClientStatsEntry OBJECT-TYPE
    SYNTAX      PintServerClientStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
     "Entries in the client server statistics table.
     One entry is defined for each client identified by name,
     monitored service type and performance statistics collection
     period."
    INDEX {pintServerClientAddress, pintServerServiceTypeIndex,
    pintServerPerfStatPeriodIndex}
    ::= { pintServerClientStatsTable 1 }

PintServerClientStatsEntry      ::=  SEQUENCE {
pintServerClientAddress                          SnmpAdminString,
pintServerClientCallsReceived                       Counter32,
pintServerClientSuccessfulCalls                     Counter32,
pintServerClientDisconnectedCalls                   Counter32,
pintServerClientDisCAutFCalls                       Counter32,
pintServerClientDisEFProbCalls                      Counter32
}

pintServerClientAddress OBJECT-TYPE
    SYNTAX      SnmpAdminString
    MAX-ACCESS not-accessible
    STATUS      current
    DESCRIPTION
     "The unique identifier of the monitored client
     identified by its address represented as as a string."
    ::= { pintServerClientStatsEntry 1 }
```

```
pintServerClientCallsReceived OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of calls received from the specific client."
    ::= { pintServerClientStatsEntry 2 }


pintServerClientSuccessfulCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of calls from the client successfully completed."
    ::= { pintServerClientStatsEntry 3 }


pintServerClientDisconnectedCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of calls received from the client, and that were
      disconnected (failed)."
    ::= { pintServerClientStatsEntry 4 }


pintServerClientDisCAutFCalls
OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS read-only
STATUS      current
DESCRIPTION
    "Number of calls from the client that were disconnected because of
    client authorization failure."
::= { pintServerClientStatsEntry 5 }


pintServerClientDisEFProbCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of calls from the client that were disconnected because
      of egress facility problems."
    ::= { pintServerClientStatsEntry 6 }


pintServerUserIdStatsTable      OBJECT-TYPE
    SYNTAX      SEQUENCE OF PintServerUserIdStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
```

```
    DESCRIPTION
     "Table displaying the monitored Pint service user statistics."
    ::= { pintServerUserIdPerf 1 }

pintServerUserIdStatsEntry OBJECT-TYPE
    SYNTAX       PintServerUserIdStatsEntry
    MAX-ACCESS  not-accessible
    STATUS       current
    DESCRIPTION
     "Entries in the user statistics table.
     One entry is defined for each user identified by name,
     each monitored service type and performance statistics collection
     period.

      It is assumed that the capabilities of the pint server
      are enough to accommodate the number of entries in this table.
      It is a local server implementation issue if an aging mechanism
      Is implemented in order to avoid scalability problems."
    INDEX {pintServerUserIdName, pintServerServiceTypeIndex,
    pintServerPerfStatPeriodIndex}
    ::= { pintServerUserIdStatsTable 1 }

PintServerUserIdStatsEntry        ::=  SEQUENCE {
pintServerUserIdName                              SnmpAdminString,
pintServerUserIdCallsReceived                      Counter32,
pintServerUserIdSuccessfulCalls                    Counter32,
pintServerUserIdDisconnectedCalls                  Counter32,
pintServerUserIdDiscUIdAFailCalls                  Counter32,
pintServerUserIdEFProbCalls                        Counter32
}

pintServerUserIdName OBJECT-TYPE
    SYNTAX      SnmpAdminString (SIZE(0..64))
    MAX-ACCESS not-accessible
    STATUS       current
    DESCRIPTION
     "The unique identifier of the monitored user
      identified by its name."
    ::= { pintServerUserIdStatsEntry 1 }

pintServerUserIdCallsReceived OBJECT-TYPE
    SYNTAX       Counter32
    MAX-ACCESS read-only
    STATUS       current
    DESCRIPTION
     "Number of calls received from the specific user."
    ::= { pintServerUserIdStatsEntry 2 }
```

```
pintServerUserIdSuccessfulCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of calls from the user successfully completed."
    ::= { pintServerUserIdStatsEntry 3 }


pintServerUserIdDisconnectedCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of calls received from the user that were
     disconnected (failed)."
    ::= { pintServerUserIdStatsEntry 4 }


pintServerUserIdDiscUIdAFailCalls
OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS read-only
STATUS      current
DESCRIPTION
    "Number of calls from the user that were disconnected because of
    user authorization failure."
::= { pintServerUserIdStatsEntry 5 }


pintServerUserIdEFProbCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of calls from the user that were disconnected because of
     egress facility problems."
    ::= { pintServerUserIdStatsEntry 6 }


pintServerGatewayStatsTable     OBJECT-TYPE
    SYNTAX       SEQUENCE OF PintServerGatewayStatsEntry
    MAX-ACCESS  not-accessible
    STATUS       current
    DESCRIPTION
     "Table displaying the monitored gateway statistics."
    ::= { pintServerGatewayPerf 1 }


pintServerGatewayStatsEntry OBJECT-TYPE
    SYNTAX       PintServerGatewayStatsEntry
    MAX-ACCESS  not-accessible
    STATUS       current
```

```
    DESCRIPTION
     "Entries in the gateway table.
      One entry is defined for each gateway identified by name,
      each monitored service type and performance statistics collection
      period."

    INDEX { pintRegisteredGatewayName, pintServerServiceTypeIndex,
    pintServerPerfStatPeriodIndex }
    ::= { pintServerGatewayStatsTable 1 }

PintServerGatewayStatsEntry      ::=  SEQUENCE {
pintServerGatewayCallsReceived                  Counter32,
pintServerGatewaySuccessfulCalls                Counter32,
pintServerGatewayDisconnectedCalls              Counter32
}

pintServerGatewayCallsReceived OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of calls received at the specified gateway."
    ::= { pintServerGatewayStatsEntry 1 }

pintServerGatewaySuccessfulCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of calls successfully completed at the specified gateway."
    ::= { pintServerGatewayStatsEntry 2 }

pintServerGatewayDisconnectedCalls OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
     "Number of calls that were disconnected (failed) at the specified
      gateway."
    ::= { pintServerGatewayStatsEntry 3 }

--
-- Notifications Section
-- (none defined)
--


--
-- Conformance Section
```

```
--

pintMibCompliances OBJECT IDENTIFIER ::= { pintMibConformance 1 }
pintMibGroups      OBJECT IDENTIFIER ::= { pintMibConformance 2 }


pintMibCompliance MODULE-COMPLIANCE
STATUS  current
DESCRIPTION
    "Describes the requirements for conformance to the
    PINT MIB."
MODULE  -- this module
MANDATORY-GROUPS { pintMibConfigGroup, pintMibMonitorGroup }
::= { pintMibCompliances 1 }


pintMibConfigGroup OBJECT-GROUP
OBJECTS {
pintReleaseNumber,
pintSysContact,
pintApplInstallPkgDescription,
pintRegisteredGatewayName,
pintRegisteredGatewayDescription
}
STATUS  current
DESCRIPTION
    "A collection of objects providing configuration
    information
    for a PINT Server."
::= { pintMibGroups 1 }


pintMibMonitorGroup OBJECT-GROUP
OBJECTS {
pintServerGlobalCallsReceived,
pintServerGlobalSuccessfulCalls,
pintServerGlobalDisconnectedCalls,
pintServerGlobalDisCUAutFCalls,
pintServerGlobalDisServProbCalls,
pintServerGlobalDisGatProbCalls,
pintServerClientCallsReceived,
pintServerClientSuccessfulCalls,
pintServerClientDisconnectedCalls,
pintServerClientDisCAutFCalls,
pintServerClientDisEFProbCalls,
--pintServerUserIdName,
pintServerUserIdCallsReceived,
pintServerUserIdSuccessfulCalls,
pintServerUserIdDisconnectedCalls,
pintServerUserIdDiscUIdAFailCalls,
pintServerUserIdEFProbCalls,
```

```
pintServerGatewayCallsReceived,
pintServerGatewaySuccessfulCalls,
pintServerGatewayDisconnectedCalls
}
STATUS   current
DESCRIPTION
     "A collection of objects providing monitoring
     information
     for a PINT Server."
::= { pintMibGroups 2 }

END
```

6. Acknowledgements

   The authors would like to thank Igor Faynberg for his encouragement
   to produce this work.

7. Security Considerations

   There is only one management object defined in this MIB that has a
   MAX-ACCESS clause of read-write (pintSysContact). There are no
   read-create objects. This read-write object 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
   information that may be sensitive from a business perspective. One
   could be the customer identification (UserIdName). Also information
   on PINT services performance might itself be need to be guarded. 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 [13] and the View-based
   Access Control Model RFC 2575 [16] 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.

8. IANA Considerations

All extensions to the values listed in this MIB must be done through
Standards Action processes as defined in RFC 2434 [20].

9. Intellectual Property

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

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

10. References

   [1]   Lu, H., Conroy, L., Bellovin, S., Krishnaswamy, M., Burg, F.,
         DeSimone, A., Tewani, K., Davidson, P., Schulzrinne, H. and K.
         Vishwanathan, "Toward the PSTN/Internet Inter-Networking --
         Pre-PINT Implementations", RFC 2458, November 1998.

   [2]   Wijnen, B., Harrington, D. and R. Presuhn, "An Architecture for
         Describing SNMP Management Frameworks", RFC 2571, April 1999.

   [3]   Rose, M. and K. McCloghrie, "Structure and Identification of
         Management Information for TCP/IP-based Internets", STD 16, RFC
         1155, May 1990.

   [4] Rose, M. and K. McCloghrie, "Concise MIB Definitions", STD 16,
        RFC 1212, March 1991.

   [5] Rose, M., "A Convention for Defining Traps for use with the
        SNMP", RFC 1215, March 1991.

   [6] McCloghrie, K., Perkins, D. and J. Schoenwaelder, "Structure of
        Management Information Version 2 (SMIv2)", STD 58, RFC 2578,
        April 1999.

   [7] McCloghrie, K., Perkins, D. and J. Schoenwaelder, "Textual
        Conventions for SMIv2", STD 58, RFC 2579, April 1999.

   [8] McCloghrie, K., Perkins, D. and J. Schoenwaelder, "Conformance
        Statements for SMIv2", STD 58, RFC 2580, April 1999.

   [9] Case, J., Fedor, M., Schoffstall, M. and J. Davin, "Simple
        Network Management Protocol", STD 15, RFC 1157, May 1990.

   [10] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser,
        "Introduction to Community-based SNMPv2", RFC 1901, January
        1996.

   [11] 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.

   [12] Case, J., Harrington D., Presuhn R. and B. Wijnen, "Message
        Processing and Dispatching for the Simple Network Management
        Protocol (SNMP)", RFC 2572, April 1999.

   [13] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM)
        for version 3 of the Simple Network Management Protocol
        (SNMPv3)", RFC 2574, April 1999.

   [14] Case, J., McCloghrie, K., Rose, M. and Waldbusser, "Protocol
        Operations for Version 2 of the Simple Network Management
        Protocol (SNMPv2)", RFC 1905, January 1996.

   [15] Levi, D., Meyer, P. and B. Stewart, "SNMPv3 Applications", RFC
        2573, April 1999.

   [16] Wijnen, B., Presuhn, R. and K. McCloghrie, "View-based Access
        Control Model (VACM) for the Simple Network Management Protocol
        (SNMP)", RFC 2575, April 1999.

   [17] Case, J., Mundy, R., Partain, D. and B. Stewart, "Introduction
        to Version 3 of the Internet-standard Network Management
        Framework", RFC 2570, April 1999.

   [18] Petrack, S. and L. Conroy, "The PINT Service Protocol:
        Extensions to SIP and SDP for IP Access to Telephone Call
        Services", RFC 2848, June 2000.

   [19] Krupczak, C. and J. Saperia, "Definitions of System-Level
        Managed Objects for Applications", RFC 2287, February 1998.

   [20] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA
        Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

11. Authors' Addresses

   Murali Krishnaswamy
   Lucent Technologies
   3C-512, 101 Crawfords Corner Rd.
   Holmdel, NJ 07733

   Phone: +1 (732)949-3611
   Fax: +1 (732)949-3210
   EMail: murali@lucent.com

   Dan Romascanu
   Avaya Communication
   Atidim Technology Park, Bldg 3
   Tel Aviv, Israel

   Phone: +972 3 6458414
   EMail: dromasca@avaya.com

12. Full Copyright Statement

Acknowledgement