Internet Draft

Andy Bierman
Cisco Systems, Inc.
21 June 2002

Capabilities MIB

<draft-ietf-sming-caps-mib-00.txt>

Status of this Memo

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

Distribution of this document is unlimited. Please send comments to the SMIng WG mailing list <sming@ops.ietf.org>.

1. Copyright Notice

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

2. 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 for identifying the conformance capabilities of all management information available from an SNMP agent.

3. Table of Contents

1 Copyright Notice 1
2 Abstract 1
3 Table of Contents 2
4 The SNMP Network Management Framework 2
5 Overview 3
5.1 MIB Structure 4
5.2 Terms 5
6 Definitions 5
7 Appendix A: Capabilities Information Usage Examples 11
7.1 Full Conformance 11
7.2 Full Conformance with some Optional Groups 11
7.3 Previous Revision Conformance 12
7.4 Partial Conformance (I) 12
7.5 Partial Conformance (II) 12
7.6 Minimum Conformance 13
7.7 Other Conformance 13
8 Acknowledgements 14
9 Open Issues 14
10 Normative References 14
11 Informative References 16
12 Security Considerations 17
13 Author's Address 17
14 Full Copyright Statement 18

4. The SNMP Network Management Framework

The SNMP Management Framework presently consists of five major components:

- An overall architecture, described in RFC 2571 [RFC2571].

- 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 RFC 1155 [RFC1155], RFC 1212 [RFC1212] and RFC 1215 [RFC1215]. The second version, called SMIv2, is described in RFC 2578 [RFC2578], RFC 2579 [RFC2579] and RFC 2580 [RFC2580].

- Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and

described in RFC 1157 [RFC1157]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in RFC 1901 [RFC1901] and RFC 1906 [RFC1906]. The third version of the message protocol is called SNMPv3 and described in RFC 1906 [RFC1906], RFC 2572 [RFC2572] and RFC 2574 [RFC2574].

- Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in RFC 1157 [RFC1157]. A second set of protocol operations and associated PDU formats is described in RFC 1905 [RFC1905].

- A set of fundamental applications described in RFC 2573 [RFC2573] and the view-based access control mechanism described in RFC 2575 [RFC2575].

A more detailed introduction to the current SNMP Management Framework can be found in RFC 2570 [RFC2570].

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.

5. Overview

There is a need for a standardized way of retrieving the conformance and variance information associated with the managed objects support by a particular agent. These management capabilities are not currently retrievable from an agent.

Conformance Statements for SMIv2 [RFC2580] provides the MODULE-COMPLIANCE macro, which allows the MIB writer to define different conformance levels, in order to specify various subsets of the full functionality defined in the MIB module, or allow for new versions of

the MIB module. Unfortunately, an NMS application developer cannot easily determine the exact conformance level by any given agent implementation, even those that are following the standard correctly. For example, it is not always easy to determine which version of a MIB module is implemented, or if an agent implements the full semantics of an object or the semantics refined by one of potentially several OBJECT clauses (in different MODULE-COMPLIANCE macros).

The AGENT-CAPABILITIES macro allows an SNMP developer to determine conformance information for specific objects for a particular release of a particular product. Unfortunately, this information is not available in any form from the agent. Instead, it is maintained as a text file by the network administrator. The sysORTable [RFC1907] allows an agent to advertise which AGENT-CAPABILITIES macros are supported, but this is insufficient because these macros are stored in external files, and since they are optional, agent developers are not required to provide these files.

5.1. MIB Structure

The Capabilities MIB contains a single group of objects, which contains a single scalar, called 'capsLastUpdateTime', and a single table, called the 'capsTable'.

The 'capsLastUpdateTime' allows an NMS application to determine if the contents of the 'capsTable' have changed over time. The 'capsTable' identifies conformance and variance capabilities information for a specified MIB sub-tree. If multiple (overlapping) entries exist for a particular portion of the MIB sub-tree, then the most specific entry (i.e. longest match) has precedence. The following objects are contained in the 'capsTable':

capsModuleCompliance
: An OID pointer to MODULE-COMPLIANCE macro pertaining to the set of objects, as implemented by the agent.

capsConfLevel
: An enumerated integer describing the basic conformance level to the specification for the set of objects, as implemented by the agent.

capsVarSyntax
: A string describing the semantic variation for a set of MIB objects or a single MIB object. This usually consists of the SYNTAX clause within the the VARIATION clause that would be present in an AGENT-CAPABILITIES macro for the specified object, or the SYNTAX, WRITE-

SYNTAX and/or MIN-ACCESS clauses that would be present in the OBJECT macro within a MODULE-COMPLIANCE macro.

5.2. Terms

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119. [RFC2119]

6. Definitions

```
CAPABILITIES-MIB DEFINITIONS ::= BEGIN

IMPORTS
        MODULE-IDENTITY, OBJECT-TYPE, mib-2
                FROM SNMPv2-SMI
        MODULE-COMPLIANCE, OBJECT-GROUP
                FROM SNMPv2-CONF
        TEXTUAL-CONVENTION, TimeStamp
                FROM SNMPv2-TC;

capabilitiesMIB MODULE-IDENTITY
    LAST-UPDATED    "200206180000Z"
    ORGANIZATION    "IETF SMIng Working Group"
    CONTACT-INFO
            "        Andy Bierman
                     Cisco Systems, Inc.
                Tel: +1 408 527-3711
             E-mail: abierman@cisco.com
             Postal: 170 West Tasman Drive
                     San Jose, CA USA 95134

             Send comments to <sming@ops.ietf.org>"
    DESCRIPTION
            "Contains objects which describe the conformance
             level capabilities for specific MIB objects."
    REVISION  "200206180000Z"
       DESCRIPTION
            "Initial version."
  ::= { mib-2 xxx }   -- unassigned


capsMibObjects       OBJECT IDENTIFIER ::= { capabilitiesMIB 1 }
capsMibNotifications OBJECT IDENTIFIER ::= { capabilitiesMIB 2 }
capsMibConformance   OBJECT IDENTIFIER ::= { capabilitiesMIB 3 }
```

```
capsMibConformanceObjects OBJECT IDENTIFIER
                                        ::= { capsMibObjects 1 }


--
-- Textual Conventions
--

ConformanceLevel ::= TEXTUAL-CONVENTION
    STATUS current
    DESCRIPTION
            "An indication of the conformance level for a particular set
            of MIB objects.

            The value 'noConf' indicates that the agent does not
            implement any of the required semantics for the associated
            set of objects.  Usually this means the object is not
            implemented at all.  However, it also indicates the special
            case in which incorrect values are returned, e.g., counter
            objects that always return zero, string objects that always
            return a zero-length string (but should not), hard-wired
            (potentially incorrect) enumerations or gauges.

            The value 'otherConf' indicates that the implementation
            complies with the semantics of the associated set of objects
            in a manner that varies from the specification in some way.

            The value 'minConf' indicates that the implementation
            complies with the semantics of the associated set of objects
            as refined by a GROUP clause or an OBJECT clause within a
            MODULE-COMPLIANCE macro.

            The value 'fullConf' indicates that the implementation
            complies with the semantics of the associated set of objects
            as defined by an indicated compliance statement for the set
            of objects.

            The value 'completeConf' indicates that the implementation
            complies with the full semantics of the associated set of
            objects as defined by the data definitions for the objects.
            This may be greater than than the conformance level
            indicated by the 'fullConf' enumeration if the indicated
            compliance statement does not require full implementation of
            the entire set of objects."
    REFERENCE
```

```
       "RFC 2580, section 5."
    SYNTAX INTEGER {
                    noConf(1),
                    otherConf(2),
                    minConf(3),
                    fullConf(4),
                    completeConf(5)
                   }


ConformanceString ::= TEXTUAL-CONVENTION
    STATUS current
    DESCRIPTION
            "An administrative string identifying conformance
            information.

            An object of this type SHOULD follow the encoding rules for
            an SnmpAdminString, except its length is limited to 1024
            octets instead of 255 octets."
    REFERENCE
       "RFC 2571, section 5."
    SYNTAX OCTET STRING (SIZE (0..1024))    -- SnmpAdminString

--
-- Conformance Group scalars
--

capsLastUpdateTime OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The value of sysUpTime when the agent last added, deleted,
            or modified an entry in the capsTable."
    ::= { capsMibConformanceObjects 1 }

--
-- Capabilities Table
--

capsTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF CapsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
```

```
            "A list of entries describing the MIB conformance
            capabilities for this agent."
    ::= { capsMibConformanceObjects 2 }

capsEntry OBJECT-TYPE
    SYNTAX      CapsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "A conceptual row in the capsTable. Entries SHOULD be
            created by the agent at system reinitialization time.
            However, agents MAY create, delete, or modify entries in
            this table to support dynamic loading and unloading of MIB
            modules. It is possible that overlapping entries will exist
            in this table, in which case the more specific entry (i.e.,
            the one with the longest capsSubTree INDEX) takes
            precedence."
    INDEX { capsSubTree }
    ::= { capsTable 1 }


CapsEntry ::= SEQUENCE {
    capsSubTree                     OBJECT IDENTIFIER,
    capsModuleCompliance            OBJECT IDENTIFIER,
    capsConfLevel                   ConformanceLevel,
    capsVarSyntax                   ConformanceString }

capsSubTree OBJECT-TYPE
    SYNTAX      OBJECT IDENTIFIER
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "The root of the MIB sub-tree associated with this entry.
            This object SHOULD identify an interior node to indicate
            multiple MIB objects or a leaf node to indicate a single MIB
            object."
    ::= { capsEntry 1 }

capsModuleCompliance OBJECT-TYPE
    SYNTAX      OBJECT IDENTIFIER
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "Identifies the MODULE-COMPLIANCE macro name for the
            associated set of MIB objects. If no appropriate value can
```

```
            be determined, then the agent will return '0.0'."
    REFERENCE
            "RFC 2580, section 5.5."
    ::= { capsEntry 2 }

capsConfLevel OBJECT-TYPE
    SYNTAX      ConformanceLevel
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "Indicates the basic level of conformance for the associated
            set of MIB objects.

            If this object contains the value 'otherConf', then the
            capsVarSyntax object SHOULD be present for the associated
            entry.

            If this object contains the value 'fullConf', 'minConf', or
            'noConf' then the capsVarSyntax object MAY be present for
            the associated entry.

            If this object contains the value 'completeConf' then the
            capsVarSyntax object SHOULD NOT be present for the
            associated entry."
    REFERENCE
            "RFC 2580, section 5."
    ::= { capsEntry 3 }

capsVarSyntax OBJECT-TYPE
    SYNTAX      ConformanceString
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "A textual representation of the syntactical variation
            described by this entry.

            If the associated capsConfLevel object is equal to
            'otherConf', then this string SHOULD contain the the same
            value as the SYNTAX clause within a VARIATION clause of an
            AGENT-CAPABILITIES macro.

            If the associated capsConfLevel object is equal to
            'minConf', and this entry identifies a set of MIB objects,
            then this string MAY identify the applicable GROUP clause(s)
            associated with the entry.
```

```
            For all entries in which the associated capsConfLevel object
            is equal to 'minConf', this string SHOULD contain the same
            value as the SYNTAX, WRITE-SYNTAX, and/or MIN-ACCESS clauses
            present within the OBJECT clause of a MODULE-COMPLIANCE
            macro. The actual keyword (e.g., 'SYNTAX') SHOULD precede
            the quoted string value.  Multiple clauses MAY be
            concatenated within a single string, and SHOULD be separated
            by whitespace.  If this entry identifies multiple objects,
            then the same clause(s) must apply to all objects.

            The agent MAY return a zero-length string instead of a
            noSuchInstance exception."
    REFERENCE
            "RFC 2580, sections 5.4.3 and 6.5.2."
    ::= { capsEntry 4 }

--
-- Notification Section  (none defined)
--

--
-- Conformance Section
--

capsMibCompliances OBJECT IDENTIFIER ::= { capsMibConformance 1 }
capsMibGroups      OBJECT IDENTIFIER ::= { capsMibConformance 2 }

capsMibCompliance MODULE-COMPLIANCE
    STATUS  current
    DESCRIPTION
            "Describes the requirements for conformance to the
            Capabilities MIB."
    MODULE  -- this module
        MANDATORY-GROUPS { capsMibConformanceGroup }

    ::= { capsMibCompliances 1 }

-- Object Groups

capsMibConformanceGroup OBJECT-GROUP
    OBJECTS {
             capsLastUpdateTime,
             capsModuleCompliance,
             capsConfLevel,
             capsVarSyntax
```

```
    }
    STATUS  current
    DESCRIPTION
            "A collection of objects used to describe the conformance
            capabilities of the MIB objects supported by an agent."
    ::= { capsMibGroups 1 }

END
```

7. Appendix A: Capabilities Information Usage Examples

The Interfaces MIB [RFC2233][RFC2863] is used to demonstrate some possible applications of the Capabilities MIB. Various Interfaces MIB capabilities are shown. Similar entries would apply to other MIB modules as well.

7.1. Full Conformance

This example shows full conformance to the RFC 2863 version of the Interfaces MIB. Note that the MODULE-COMPLIANCE allows many objects to be omitted, as specified by the various GROUP macros. The enumeration 'fullConf' indicates that the full implementation of all mandatory objects is supported, even if OBJECT clauses exist for some mandatory objects.

```
 - capsTable entry for the entire IF-MIB:
    - capsSubTree INDEX == { ifMIB 1 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'fullConf(4)'
    - capsVarSyntax == noSuchInstance (or empty string)
```

7.2. Full Conformance with some Optional Groups

This example shows full conformance to the RFC 2863 version of the Interfaces MIB. In this example, the agent supports the groups required for character-oriented interfaces with ifSpeed values not exceeding 20,000,000 bits/second.

```
 - capsTable entry for the entire IF-MIB:
    - capsSubTree INDEX == { ifMIB 1 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'fullConf(4)'
    - capsVarSyntax == 'GROUP ifFixedLengthGroup'
```

7.3. Previous Revision Conformance

This example shows full conformance to the RFC 2233 version of the Interfaces MIB. In this example, the agent supports all the mandatory groups, and supports the group for high capacity packet based interfaces.

```
 - capsTable entry for the entire IF-MIB:
    - capsSubTree INDEX == { ifMIB 1 }
    - capsModuleCompliance == { ifCompliance2 }
    - capsConfLevel == 'fullConf(4)'
    - capsVarSyntax == 'GROUP ifHCPacketGroup'
```

7.4. Partial Conformance (I)

This example shows complete conformance to the RFC 2863 version of the Interfaces MIB. In this example, the agent supports all the mandatory groups and all the optional groups, however, the deprecated ifTestTable is not implemented.

```
 - capsTable entry for the entire IF-MIB:
    - capsSubTree INDEX == { ifMIB 1 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'completeConf(5)'
    - capsVarSyntax == noSuchInstance (or empty string)

 - capsTable entry for the ifTestTable:
    - capsSubTree INDEX == { ifMIBObjects 3 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'noConf(1)'
    - capsVarSyntax == noSuchInstance (or empty string)
```

7.5. Partial Conformance (II)

This example shows full conformance to the RFC 2863 version of the ifTable and ifXTable, except for the ifTable non-unicast packet counters, which are not implemented at all.

```
 - capsTable entry for the ifTable:
    - capsSubTree INDEX == { interfaces 2 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'fullConf(4)'
    - capsVarSyntax == noSuchInstance (or empty string)

 - capsTable entry for the ifXTable:
```

```
    - capsSubTree INDEX == { ifMIBObjects 1 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'fullConf(4)'
    - capsVarSyntax == noSuchInstance (or empty string)

 - capsTable entry for the ifInNUcastPkts:
    - capsSubTree INDEX == { ifEntry 12 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'noConf(1)'
    - capsVarSyntax == noSuchInstance (or empty string)

 - capsTable entry for the ifOutNUcastPkts:
    - capsSubTree INDEX == { ifEntry 18 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'noConf(1)'
    - capsVarSyntax == noSuchInstance (or empty string)
```

7.6. Minimum Conformance

This example shows minimum conformance to the RFC 2863 version of the ifTable. In this example, the agent supports all the mandatory objects, except the table is implemented as read-only.

```
 - capsTable entry for the ifTable:
    - capsSubTree INDEX == { interfaces 2 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'minConf(3)'
    - capsVarSyntax == 'MIN-ACCESS read-only'
```

7.7. Other Conformance

This example shows full conformance to the RFC 2863 version of the ifXTable, except the ifAlias object, which is implemented as a read-write 32 octet string, instead of a read-write 64 octet string.

```
 - capsTable entry for the ifXTable:
    - capsSubTree INDEX == { ifMIBObjects 1 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'fullConf(4)'
    - capsVarSyntax == noSuchInstance (or empty string)

 - capsTable entry for ifAlias:
    - capsSubTree INDEX == { ifXEntry 18 }
    - capsModuleCompliance == { ifCompliance3 }
    - capsConfLevel == 'otherConf(2)'
```

- capsVarSyntax == 'SYNTAX DisplayString (SIZE(0..32))'

8. Acknowledgements

This memo is a product of the SMIng working group.

9. Open Issues

Conformance information for INDEX components
: Should the capsTable be modified to support entries for INDEX components, which are not actually represented in MODULE-COMPLIANCE macros?

Default capsTable behavior
: Should the absence of a capsEntry indicate any particular semantics?

Missing conformance sections
: MODULE-COMPLIANCE sections are not strictly mandatory, but this MIB gets around this corner-case (and the missing conformance information for INDEX components) by describing the conformance information that would be present in an information module, even if that information is not actually present. Are there any problems with this approach?

Standardized Entry Format
: There are many different ways to represent the same level of support for a given set of MIB objects in the capsTable. Should there be standard guidelines for populating the capsTable for various usage scenerios?

10. Normative References

[RFC1905]
: SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1905, SNMP Research, Inc., Cisco Systems, Inc., Dover Beach Consulting, Inc., International Network Services, January 1996.

[RFC1906]
: SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1906, SNMP Research, Inc., Cisco Systems, Inc., Dover Beach Consulting, Inc., International Network

Services, January 1996.

[RFC2026]
Bradner, S., "The Internet Standards Process -- Revision 3", RFC 2026, Harvard University, October, 1996.

[RFC2119]
S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels" RFC 2119, Harvard University, March 1997.

[RFC2571]
Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing SNMP Management Frameworks", RFC 2571, Cabletron Systems, Inc., BMC Software, Inc., IBM T. J. Watson Research, April 1999.

[RFC2572]
Case, J., Harrington D., Presuhn R., and B. Wijnen, "Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)", RFC 2572, SNMP Research, Inc., Cabletron Systems, Inc., BMC Software, Inc., IBM T. J. Watson Research, April 1999.

[RFC2573]
Levi, D., Meyer, P., and B. Stewart, "SNMPv3 Applications", RFC 2573, SNMP Research, Inc., Secure Computing Corporation, Cisco Systems, April 1999.

[RFC2574]
Blumenthal, U., and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", RFC 2574, IBM T. J. Watson Research, April 1999.

[RFC2575]
Wijnen, B., Presuhn, R., and K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)", RFC 2575, IBM T. J. Watson Research, BMC Software, Inc., Cisco Systems, Inc., April 1999.

[RFC2578]
McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", RFC 2578, STD 58, Cisco Systems, SNMPinfo, TU Braunschweig, SNMP Research, First Virtual Holdings, International Network Services, April 1999.

[RFC2579]
McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Textual Conventions for SMIv2", RFC 2579, STD 58, Cisco Systems, SNMPinfo, TU Braunschweig, SNMP Research, First Virtual Holdings, International Network Services, April 1999.

[RFC2580]
McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Conformance Statements for SMIv2", RFC 2580, STD 58, Cisco Systems, SNMPinfo, TU Braunschweig, SNMP Research, First Virtual Holdings, International Network Services, April 1999.

11. Informative References

[RFC1155]
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.

[RFC1157]
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.

[RFC1212]
Rose, M., and K. McCloghrie, "Concise MIB Definitions", RFC 1212, Performance Systems International, Hughes LAN Systems, March 1991.

[RFC1215]
M. Rose, "A Convention for Defining Traps for use with the SNMP", RFC 1215, Performance Systems International, March 1991.

[RFC1901]
SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Introduction to Community-based SNMPv2", RFC 1901, SNMP Research, Inc., Cisco Systems, Inc., Dover Beach Consulting, Inc., International Network Services, January 1996.

[RFC1907]
SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1907, SNMP Research, Inc., Cisco Systems, Inc., Dover Beach Consulting, Inc., International Network Services, January 1996.

[RFC2233]
McCloghrie, K., and F. Kastenholz, "The Interfaces Group MIB Using SMIv2", RFC 2233, Cisco Systems, FTP Software, November, 1997.

[RFC2570]
Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction to Version 3 of the Internet-standard Network Management Framework", RFC 2570, SNMP Research, Inc., TIS Labs at Network Associates, Inc., Ericsson, Cisco Systems, April 1999.

[RFC2863]
McCloghrie, K., and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, Cisco Systems, Argon Networks, June, 2000.

12. Security Considerations

There are no managed objects defined in this MIB that have a MAX-ACCESS clause of read-write and/or read-create.

There are no managed objects in this MIB that may contain sensitive information.

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 the objects in this MIB.

It is recommended that the implementors consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model RFC 2574 [RFC2574] and the View-based Access Control Model RFC 2575 [RFC2575] 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 GET them.

13. Author's Address

Andy Bierman
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA USA 95134
Phone: +1 408-527-3711
Email: abierman@cisco.com

14. 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.