

Internet Engineering Task Force (IETF)
INTERNET DRAFT: <[draft-white-slapm-mib-00.txt](#)>
Expiration Date: February 1999

Kenneth White
IBM Corp.

August 1998

**Definitions of Managed Objects for
Service Level Agreements
Performance Monitoring**
<[draft-white-slapm-mib-00.txt](#)>

Status of this Memo

This document is an Internet Draft. 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. Internet Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material or to cite them other than as a "working draft" or "work in progress."

Please check the I-D abstract listing contained in each Internet Draft directory to learn the current status of this or any Internet Draft. Distribution of this document is unlimited.

Copyright Notice

Copyright (C) The Internet Society (1998). All Rights Reserved.

Abstract

This memo defines a Management Information Base (MIB) for performing performance monitoring of Service Level Agreements (SLAs) defined via policy definitions. The MIB defined herein focuses on defining a set of objects for monitoring SLAs and not on replication of the content of policy definitions.

The MIB defined by this document is based upon the content of "Schema for Service Level Administration of Differential Services and Integrated Services in Networks", refer to [19].

Table of Contents

1.0	Introduction	2
---------------------	------------------------	-------------------

2.0	The SNMP Network Management Framework	2
3.0	Structure of the MIB	3
3.1	Global simple objects	4
3.2	slapmPolicyStatsTable	4
3.3	slapmPolicyMonitorTable	4
3.4	slapmTcpConnTable and slapmUdpTable	5
4.0	Definitions	5
5.0	Security Considerations	29
6.0	Intellectual Property	29
7.0	Acknowledgments	29
8.0	References	29
9.0	Author's Address	31
10.0	Full Copyright Statement	31

[1.0](#) Introduction

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](#), reference [13].

This document is a product of the Internet Engineering Task Force (IETF). Its purpose is to define a MIB module for performance management of Service Level Agreements (SLAs). An SLA is defined via policy. There are a number of methods that exist for how policy is defined and administered. Definition of these methods is considered outside of the scope of this document. For modeling the contents of this MIB a policy definition is considered to consist of a set of profiles that select the conditions for when a policy action should be applied. Refer to "Schema for Service Level Administration of Differential Services and Integrated Services in Networks" [19] for a definition of policy, traffic profile and action.

The SLAPM-MIB is structured primarily to enable monitoring policy actions at a system. A system can be either an edge device (end-system) or an interior device or node (e.g. router).

[2.0](#) The SNMP Network Management Framework

The SNMP Management Framework presently consists of five major

components:

- o An overall architecture, described in [RFC 2271](#) [7].

- 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 [RFC 1155](#) [14], [RFC 1212](#) [15] and [RFC 1215](#) [16]. The second version, called SMIV2, is described in [RFC 1902](#) [3], [RFC 1903](#) [4] and [RFC 1904](#) [5].
- o Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and described in [RFC 1157](#) [1]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in [RFC 1901](#) [17] and [RFC 1906](#) [18]. The third version of the message protocol is called SNMPv3 and described in [RFC 1906](#) [18], [RFC 2272](#) [8] and [RFC 2274](#) [10].
- o Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in [RFC 1157](#) [1]. A second set of protocol operations and associated PDU formats is described in [RFC 1905](#) [6].
- o A set of fundamental applications described in [RFC 2273](#) [9] and the view-based access control mechanism described in [RFC 2275](#) [11].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined ore, 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.

[3.0](#) Structure of the MIB

The SLAPM-MIB consists of the following components:

- o Global simple objects
- o slapmPolicyStatsTable
- o slapmPolicyMonitorTable
- o slapmTcpConnTable and slapmUdpTable

Refer to the compliance statement defined within SLAPM-MIB for a definition of what objects MUST be implemented by all systems as opposed to those that MUST be implemented by host systems only.

White, Kenneth

Expires December 1998

[Page 3]

3.1 Global simple objects

Global objects defined within SLAPM-MIB:

- o `slapmSpinLock`

Enables multiple management application access to SLAPM-MIB. An agent **MUST** implement the `slapmSpinLock` object to enable management applications to coordinate their use of the SLAPM-MIB. Management application use of `slapmSpinLock` is **OPTIONAL**.

- o `slapmPolicyQueryCount`, `slapmPolicyAccessCount`,
`slapmPolicySuccessCount`, and `slapmPolicyNotFoundCount`

Basic statistics on the amount of policy directory access that has occurred at a system.

- o `slapmPolicyPurgeTime`

Used to prevent the entries in various SLAPM-MIB tables that relate to a policy definition from immediately being deleted when the corresponding policy definition no longer exists. This gives management applications time to discover this condition and close out any polled based interval data that may be being collected. Refer to the OBJECT description for `slapmPolicyPurgeTime` for a more precise description of this function.

3.2 `slapmPolicyStatsTable`

The `slapmPolicyStatsTable` is the main table defined by SLAPM-MIB. One `slapmPolicyStatsEntry` exists for each SLA traffic profile per policy definition. The structure of indexing for `slapmPolicyStatsTable` assumes that it is permissible for a policy to consist of a set of traffic profiles. Systems that allow only a single traffic profile to be specified per policy definition should still use the name of the traffic profile as the second index into `slapmPolicyStatsTable` in order to conform to SLAPM-MIB. Use of a zero-length octet string to indicate this condition is **NOT** allowed.

SLAPM-MIB also assumes that only a single action can be defined for any single policy definition and hence an action's name doesn't need to be reflected in table indexing. This table provides basic statistics on the set of policy traffic profiles known at a system. Entries in this table are not administered via SNMP. An agent implementation for this table **MUST** reflect its current set of policy definitions via table entries. The mechanisms for policy administration are outside of the scope of this memo.

3.3 `slapmPolicyMonitorTable`

The slapmPolicyMonitorTable provides a method of monitoring the effect of SLA policy being used at a system. A management application creates an slapmPolicyMonitorEntry for each collection that it requires. The

value of the BITS `slapmPolicyMonitorControl` object determines what type of collection occurs:

o `monitorMinRate(0)`

Use the value of `slapmPolicyMonitorInterval` as the interval to determine current traffic rate, `slapmPolicyMonitorCurrentInRate` plus `slapmPolicyMonitorCurrentOutRate`, that can be compared to `slapmPolicyMonitorMinRateLow` for determining when to generate a `slapmMinRateNotAchieved` notification. The notification `slapmMinRateOkay` is generated when the problem is resolved. This can be determined by comparing the current rate to `slapmPolicyMonitorMinRateHigh`.

o `monitorMaxRate(1)`

Use the value of `slapmPolicyMonitorInterval` as the interval to determine current traffic rate, `slapmPolicyMonitorCurrentInRate` plus `slapmPolicyMonitorCurrentOutRate`, that can be compared to `slapmPolicyMonitorMaxRateHigh` for determining when to generate a `slapmMaxRateExceeded` notification. The notification `slapmMaxRateOkay` is generated when the problem is resolved. This can be determined by comparing the current rate to `slapmPolicyMonitorMaxRateLow`.

o `monitorMaxDelay(2)`

Use the value of `slapmPolicyMonitorInterval` as the interval to determine the current delay. This can be calculated by averaging the round trip times for all TCP connections associated with the policy definition. Compare this value to `slapmPolicyMonitorMaxDelayHigh` for determining when to generate a `slapmMaxDelayExceeded` notification. The notification `slapmMaxDelayOkay` is generated when the problem is resolved. This can be determined by comparing the current rate to `slapmPolicyMonitorMaxDelayLow`.

The index element `slapmPolicyMonitorOwnerIndex` is used as the first index in `slapmPolicyMonitorTable` in order to enable SNMPv3 VACM security control. The `slapmPolicyMonitorTable` is the only table that supports SNMP RowStatus operations.

3.4 `slapmTcpConnTable` and `slapmUdpTable`

Entries are made into `slapmTcpConnTable` or `slapmUdpTable` to indicate actual policy usage and to provide general statistics on a TCP connection or UDP listener level.

4.0 Definitions

White, Kenneth

Expires December 1998

[Page 5]

SLAPM-MIB DEFINITIONS ::= BEGIN

IMPORTS

MODULE-IDENTITY, OBJECT-TYPE,
experimental, Unsigned32, NOTIFICATION-TYPE,
Gauge32, Counter32, TimeTicks
FROM SNMPv2-SMI -- [RFC1902](#)
TEXTUAL-CONVENTION, RowStatus,
TestAndIncr, DateAndTime
FROM SNMPv2-TC -- [RFC1903](#)
MODULE-COMPLIANCE, OBJECT-GROUP,
NOTIFICATION-GROUP
FROM SNMPv2-CONF -- [RFC1904](#)
SnmAdminString
FROM SNMP-FRAMEWORK-MIB; -- [RFC2271](#)

slapmMIB MODULE-IDENTITY

LAST-UPDATED "9808190000Z"
ORGANIZATION "Internet Engineering Task Force (IETF)"
CONTACT-INFO
"Kenneth White

International Business Machines Corporation
Network Computing Software Division
Research Triangle Park, NC, USA

E-mail: kennethw@vnet.ibm.com"

DESCRIPTION

"The Service Level Agreement Performance Monitoring MIB
(SLAPM-MIB) provides data collection and monitoring
capabilities for the Service Level Agreements (SLAs)
policy definitions."

::= { experimental 2001 } -- Need real IANA experimental OID

-- Textual Conventions

SlapmNameType ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"The textual convention for naming entities
within this MIB. The actual contents of an object
defined using this textual convention should consist
of the distinguished name portion of a name.
This is usually the right-most
portion of the name. This convention is necessary,
since names within this MIB can be used as index
items and an instance identifier is limited to 128
subidentifiers."

SYNTAX SnmpAdminString (SIZE(0..32))

-- Top-level structure of the MIB

slapmNotifications OBJECT IDENTIFIER ::= { slapmMIB 0 }

White, Kenneth

Expires December 1998

[Page 6]

```
slapmObjects      OBJECT IDENTIFIER ::= { slapmMIB 1 }
slapmConformance  OBJECT IDENTIFIER ::= { slapmMIB 2 }
```

-- All simple objects

```
slapmBaseObjects  OBJECT IDENTIFIER ::= { slapmObjects 1 }
```

-- Simple Object Definitions

```
slapmSpinLock OBJECT-TYPE
```

```
    SYNTAX      TestAndIncr
```

```
    MAX-ACCESS  read-write
```

```
    STATUS      current
```

```
    DESCRIPTION
```

"An advisory lock used to allow cooperating applications to coordinate their use of the contents of this MIB. This typically occurs when an application seeks to create a new entry or alter an existing entry in slapmPolicyMonitorTable. A management implementation MAY utilize the slapmSpinLock to serialize its changes or additions. This usage is not required. However, slapmSpinLock MUST be supported by agent implementations."

```
    ::= { slapmBaseObjects 1 }
```

```
slapmPolicyQueryCount OBJECT-TYPE
```

```
    SYNTAX      Counter32
```

```
    MAX-ACCESS  read-only
```

```
    STATUS      current
```

```
    DESCRIPTION
```

"The total number of times that a policy lookup occurred. This is the number of times that a reference was made to a policy definition at a system and includes the number of times that a policy repository was accessed, slapmPolicyAccessCount. The object slapmPolicyAccessCount should be less than slapmPolicyQueryCount when policy definitions are cached at a system."

```
    ::= { slapmBaseObjects 2 }
```

```
slapmPolicyAccessCount OBJECT-TYPE
```

```
    SYNTAX      Counter32
```

```
    MAX-ACCESS  read-only
```

```
    STATUS      current
```

```
    DESCRIPTION
```

"Total number of times that a policy repository was accessed. The value of this object should be less than slapmPolicyQueryCount, since typically policy entries are cached to minimize repository accesses."

```
    ::= { slapmBaseObjects 3 }
```

slapmPolicySuccessAccessCount OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

White, Kenneth

Expires December 1998

[Page 7]

DESCRIPTION

"Total number of successful policy repository accesses."

::= { slapmBaseObjects 4 }

slapmPolicyNotFoundCount OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of policy repository accesses that resulted in an entry not being located."

::= { slapmBaseObjects 5 }

slapmPolicyPurgeTime OBJECT-TYPE

SYNTAX Unsigned32 (0..3600) -- maximum of 1 hour

UNITS "seconds"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The purpose of this object is to define the amount of time (in seconds) to wait before removing an slapmPolicyStatsEntry when a system detects that the associated policy definition has been deleted. This gives any polling management applications time to complete their last poll before an entry is removed. An slapmPolicyStatsEntry enters the deleteNeeded(3) state via slapmPolicyStatsOperStatus when a system first detects that the entry needs to be removed.

Once an slapmPolicyStatsEntry has becomes eligible for deletion (slapmPolicyPurgeTime has expired) it is removed as long as there are no dependent entries in the slapmPolicyMonitorTable. An slapmPolicyStatsEntry enters the deleted(4) state as defined by slapmPolicyStatsOperstatus when it needs to be deleted but can't due to dependent slapmPolicyMonitorTable entries. A slapmPolicyDeleteNeeded notification is generated when this condition is detected.

A value of 0 for this option disables this function and results in either the purging of slapmPolicyTable entries upon transition into deleteNeeded(3) state or generation of a slapmPolicyDeleteNeeded notification."

DEFVAL { 900 } -- 15 minute default purge time

::= { slapmBaseObjects 6 }

-- Sla Performance Monitoring Policy Statistics Table

slapmPolicyStatsTable OBJECT-TYPE
SYNTAX SEQUENCE OF SlapmPolicyStatsEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION

White, Kenneth

Expires December 1998

[Page 8]

"Provides statistics on all policies known at a system."
 ::= { slapmObjects 2 }

slapmPolicyStatsEntry OBJECT-TYPE

SYNTAX SlapmPolicyStatsEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Defines an entry in the slapmPolicySetupTable. This table defines a set of statistics that is kept on a per policy and traffic profile basis. A policy can be defined to contain multiple traffic profiles that map to a single action.

Entries in this table are not created or deleted via SNMP but reflect the set of policy definitions known at a system."

INDEX {
 slapmPolicyStatsPolicyName,
 slapmPolicyStatsTrafficProfileName
}

::= { slapmPolicyStatsTable 1 }

SlapmPolicyStatsEntry ::=

SEQUENCE {
 slapmPolicyStatsPolicyName SlapmNameType,
 slapmPolicyStatsTrafficProfileName SlapmNameType,
 slapmPolicyStatsOperStatus INTEGER,
 slapmPolicyStatsActiveConns Gauge32,
 slapmPolicyStatsFirstActivated DateAndTime,
 slapmPolicyStatsLastMapping DateAndTime,
 slapmPolicyStatsLastActivity DateAndTime,
 slapmPolicyStatsInOctets Counter32,
 slapmPolicyStatsOutOctets Counter32,
 slapmPolicyStatsConnectionLimit Unsigned32,
 slapmPolicyStatsActiveConnections Gauge32,
 slapmPolicyStatsAcceptCount Counter32,
 slapmPolicyStatsDenyCount Counter32
}

slapmPolicyStatsPolicyName OBJECT-TYPE

SYNTAX SlapmNameType

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Policy name that this entry relates to."

::= { slapmPolicyStatsEntry 1 }

slapmPolicyStatsTrafficProfileName OBJECT-TYPE

SYNTAX SlapmNameType

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The name of a traffic profile that is associated with a policy."

::= { slapmPolicyStatsEntry 2 }

White, Kenneth

Expires December 1998

[Page 9]

slapmPolicyStatsOperStatus OBJECT-TYPE

SYNTAX INTEGER {
inactive(1),
active(2),
deleteNeeded(3),
deleted(4)
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The state of a policy entry:

- inactive(1) - An policy entry was either defined by local SYSDEF or discovered via a directory search but has not been activated (not currently being used).
- active(2) - Policy entry is being used to affect traffic flows.
- deleteNeeded(3) - Either through local implementation dependent methods or by discovering that the directory entry corresponding to this table entry no longer exists and slapmPolicyPurgeTime needs to expire before attempting to remove the corresponding slapmPolicyStatsEntry.
- deleted(4) - Entry needs to be removed but can't due to dependent slapmPolicyMonitorTable entries existing."

::= { slapmPolicyStatsEntry 3 }

slapmPolicyStatsActiveConns OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of active connects that are affected by the corresponding policy entry."

::= { slapmPolicyStatsEntry 4 }

slapmPolicyStatsFirstActivated OBJECT-TYPE

SYNTAX DateAndTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The timestamp for when the corresponding policy entry is activated. The value of this object serves as the discontinuity event indicator when polling entries in this table. The value of this object is updated on

```
transition of slapmPolicyStatsOperStatus into the active(2)
state."
::= { slapmPolicyStatsEntry 5 }
```

slapmPolicyStatsLastMapping OBJECT-TYPE

White, Kenneth

Expires December 1998

[Page 10]

SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "The timestamp for when the last time
 that the associated policy entry was used."
::= { slapiPolicyStatsEntry 6 }

slapiPolicyStatsLastActivity OBJECT-TYPE

SYNTAX DateAndTime
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "The timestamp for the last time that traffic occurred for
 entities that map to this entry."
::= { slapiPolicyStatsEntry 7 }

slapiPolicyStatsInOctets OBJECT-TYPE

SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "The number of octets that was received by IP for an entity
 that map to this entry."
::= { slapiPolicyStatsEntry 8 }

slapiPolicyStatsOutOctets OBJECT-TYPE

SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "The number of octets that was transmitted by IP for an entity
 that map to this entry."
::= { slapiPolicyStatsEntry 9 }

slapiPolicyStatsConnectionLimit OBJECT-TYPE

SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "The limit for the number of active connections that are
 allowed for this policy definition."
::= { slapiPolicyStatsEntry 10 }

slapiPolicyStatsActiveConnections OBJECT-TYPE

SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"The number of active connections that mapped to the
corresponding policy definition."
::= { slapmPolicyStatsEntry 11 }

slapmPolicyStatsAcceptCount OBJECT-TYPE

White, Kenneth

Expires December 1998

[Page 11]

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "This counter is incremented when a policy action's
    Permission value is set to Accept and a session is accepted."
 ::= { slapmPolicyStatsEntry 12 }
```

slapmPolicyStatsDenyCount OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "This counter is incremented when a policy action's
    Permission value is set to Deny and a session is denied,
    or when a session is rejected due to a policy's
    connection limit (slapmPolicyStatsConnectLimit) being
    reached."
 ::= { slapmPolicyStatsEntry 13 }
```

-- SLA Performance Monitoring Policy Monitor Table

slapmPolicyMonitorTable OBJECT-TYPE

```
SYNTAX SEQUENCE OF SlapmPolicyMonitorEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "Provides a method of monitoring policies at a
    system."
 ::= { slapmObjects 3 }
```

slapmPolicyMonitorEntry OBJECT-TYPE

```
SYNTAX SlapmPolicyMonitorEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "Defines an entry in the slapmPolicyMonitorTable. This table
    defines which policies should be monitored on a per policy
    action basis."
INDEX {
    slapmPolicyMonitorOwnerIndex,
    slapmPolicyMonitorPolicyName,
    slapmPolicyMonitorTrafficProfileName
}
 ::= { slapmPolicyMonitorTable 1 }
```

SlapmPolicyMonitorEntry ::=

```
SEQUENCE {
```

slapmPolicyMonitorOwnerIndex
slapmPolicyMonitorPolicyName
slapmPolicyMonitorTrafficProfileName
slapmPolicyMonitorControl
slapmPolicyMonitorInterval

SnmpAdminString,
SlapmNameType,
SlapmNameType,
BITS,
Unsigned32,

White, Kenneth

Expires December 1998

[Page 12]

slapmPolicyMonitorIntTime	DateAndTime,
slapmPolicyMonitorCurrentInRate	Gauge32,
slapmPolicyMonitorCurrentOutRate	Gauge32,
slapmPolicyMonitorMinRateLow	Unsigned32,
slapmPolicyMonitorMinRateHigh	Unsigned32,
slapmPolicyMonitorMaxRateHigh	Unsigned32,
slapmPolicyMonitorMaxRateLow	Unsigned32,
slapmPolicyMonitorMaxDelayHigh	Unsigned32,
slapmPolicyMonitorMaxDelayLow	Unsigned32,
slapmPolicyMonitorMinRateNotAchieved	Counter32,
slapmPolicyMonitorMaxRateExceeded	Counter32,
slapmPolicyMonitorMaxDelayExceeded	Counter32,
slapmPolicyMonitorRowStatus	RowStatus

}

slapmPolicyMonitorOwnerIndex OBJECT-TYPE

SYNTAX SnmpAdminString (SIZE(0..16))

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"To facilitate the provisioning of access control by a security administrator using the View-Based Access Control Model ([RFC 2275](#), VACM) for tables in which multiple users may need to independently create or modify entries, the initial index is used as an 'owner index'. Such an initial index has a syntax of SnmpAdminString, and can thus be trivially mapped to a securityName or groupName as defined in VACM, in accordance with a security policy.

All entries in that table belonging to a particular user will have the same value for this initial index. For a given user's entries in a particular table, the object identifiers for the information in these entries will have the same subidentifiers (except for the 'column' subidentifier) up to the end of the encoded owner index. To configure VACM to permit access to this portion of the table, one would create vacmViewTreeFamilyTable entries with the value of vacmViewTreeFamilySubtree including the owner index portion, and vacmViewTreeFamilyMask 'wildcarding' the column subidentifier. More elaborate configurations are possible."

```
::= { slapmPolicyMonitorEntry 1 }
```

slapmPolicyMonitorPolicyName OBJECT-TYPE

SYNTAX SlapmNameType

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Policy name that this entry relates to."

```
::= { slapmPolicyMonitorEntry 2 }
```

slapmPolicyMonitorTrafficProfileName OBJECT-TYPE

SYNTAX SlapmNameType
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "The corresponding Traffic Profile name."

White, Kenneth

Expires December 1998

[Page 13]

```
::= { slapmPolicyMonitorEntry 3 }
```

```
slapmPolicyMonitorControl OBJECT-TYPE
```

```
SYNTAX      BITS {
                monitorMinRate(0),
                monitorMaxRate(1),
                monitorMaxDelay(2)
            }
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The value of this object determines the type of monitoring
that is applied to a policy/action pair.  The value of this
object can't be changed once the table entry that it is a
part of is activated via slapmPolicyMonitorRowStatus."
```

```
DEFVAL      { { monitorMinRate, monitorMaxRate } }
```

```
::= { slapmPolicyMonitorEntry 4 }
```

```
slapmPolicyMonitorInterval OBJECT-TYPE
```

```
SYNTAX      Unsigned32 (15..86400) -- 15 second min, 24 hour max
```

```
UNITS       "seconds"
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The number of seconds that defines the sample period."
```

```
DEFVAL      {20} -- 20 seconds
```

```
::= { slapmPolicyMonitorEntry 5 }
```

```
slapmPolicyMonitorIntTime OBJECT-TYPE
```

```
SYNTAX      DateAndTime
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The timestamp for when the last interval ended."
```

```
::= { slapmPolicyMonitorEntry 6 }
```

```
slapmPolicyMonitorCurrentInRate OBJECT-TYPE
```

```
SYNTAX      Gauge32
```

```
UNITS       "kilobits per second"
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Using the value of the corresponding slapmPolicyMonitorInterval,
slapmPolicyStatsInOctets is sampled and then divided by
slapmPolicyMonitorInterval to determine the current in transfer
rate."
```

```
::= { slapmPolicyMonitorEntry 7 }
```

```
slapmPolicyMonitorCurrentOutRate OBJECT-TYPE
```

SYNTAX	Gauge32
UNITS	"kilobits per second"
MAX-ACCESS	read-only
STATUS	current
DESCRIPTION	

White, Kenneth

Expires December 1998

[Page 14]

"Using the value of the corresponding slapmPolicyMonitorInterval, slapmPolicyStatsOutOctets is sampled and then divided by slapmPolicyMonitorInterval to determine the current out transfer rate."

::= { slapmPolicyMonitorEntry 8 }

slapmPolicyMonitorMinRateLow OBJECT-TYPE

SYNTAX Unsigned32

UNITS "kilobits per second"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The threshold for generating a slapmMinRateNotAchieved notification, signalling that a monitored minimum transfer rate has not been meet. The objects slapmPolicyMonitorCurrentInRate and slapmPolicyMonitorCurrentOutRate are summed when compared to the value of this object.

A slapmMinRateNotAchieved notification is not generated again for an slapmPolicyMonitorEntry until the minimum transfer rate exceeds slapmPolicyMonitorMinRateHigh (a slapmMinRateOkay notification is then transmitted) and then fails below slapmPolicyMonitorMinRateLow. This behavior reduces the slapmMinRateNotAchieved notifications that are transmitted.

A value of zero for this object is returned when the slapmPolicyMonitorControl monitorMinRate(0) is not enabled. When enabled the default value for this object is the min rate value specified in the associated action definition minus 10%. If the action definition doesn't have a min rate defined then there is no default for this object and a value MUST be specified prior to activating this entry when monitorMinRate(0) is selected."

::= { slapmPolicyMonitorEntry 9 }

slapmPolicyMonitorMinRateHigh OBJECT-TYPE

SYNTAX Unsigned32

UNITS "kilobits per second"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The threshold for generating a slapmMinRateOkay notification, signalling that a monitored minimum transfer rate has increased to an acceptable level.

A value of zero for this object is returned when the slapmPolicyMonitorControl monitorMinRate(0) is not enabled. When enabled the default value for this object

is the min rate value specified in the associated action definition plus 10%. If the action definition doesn't have a min rate defined then there is no default for this object and a value MUST be specified prior to activating this entry when monitorMinRate(0)

White, Kenneth

Expires December 1998

[Page 15]

```
is selected."  
 ::= { slapmPolicyMonitorEntry 10 }
```

slapmPolicyMonitorMaxRateHigh OBJECT-TYPE

SYNTAX Unsigned32

UNITS "kilobits per second"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The threshold for generating a slapmMaxRateExceeded notification, signalling that a monitored maximum transfer rate has been exceeded. The objects slapmPolicyMonitorCurrentInRate and slapmPolicyMonitorCurrentOutRate are summed when compared to the value of this object.

A slapmMaxRateExceeded notification is not generated again for an slapmPolicyMonitorEntry until the maximum transfer rate fails below slapmPolicyMonitorMaxRateLow (a slapmMaxRateOkay notification is then transmitted) and then raises above slapmPolicyMonitorMaxRateHigh. This behavior reduces the slapmMaxRateExceeded notifications that are transmitted.

A value of zero for this object is returned when the slapmPolicyMonitorControl monitorMaxRate(1) is not enabled. When enabled the default value for this object is the max rate value specified in the associated action definition plus 10%. If the action definition doesn't have a max rate defined then there is no default for this object and a value MUST be specified prior to activating this entry when monitorMaxRate(1) is selected."

```
 ::= { slapmPolicyMonitorEntry 11 }
```

slapmPolicyMonitorMaxRateLow OBJECT-TYPE

SYNTAX Unsigned32

UNITS "kilobits per second"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The threshold for generating a slapmMaxRateOkay notification, signalling that a monitored maximum transfer rate has fallen to an acceptable level.

A value of zero for this object is returned when the slapmPolicyMonitorControl monitorMaxRate(1) is not enabled. When enabled the default value for this object is the max rate value specified in the associated action definition minus 10%. If the action definition doesn't have a max rate defined then there is no

```
default for this object and a value MUST be specified
prior to activating this entry when monitorMaxRate(1)
is selected."
::= { slapmPolicyMonitorEntry 12 }
```


slapmPolicyMonitorMaxDelayHigh OBJECT-TYPE

SYNTAX Unsigned32
UNITS "milliseconds"
MAX-ACCESS read-create
STATUS current

DESCRIPTION

"The threshold for generating a slapmMaxDelayExceeded notification, signalling that a monitored maximum delay rate has been exceeded.

A slapmMaxDelayExceeded notification is not generated again for an slapmPolicyMonitorEntry until the maximum delay rate falls below slapmPolicyMonitorMaxDelayLow (a slapmMaxDelayOkay notification is then transmitted) and raises above slapmPolicyMonitorMaxDelayHigh. This behavior reduces the slapmMaxDelayExceeded notifications that are transmitted.

A value of zero for this object is returned when the slapmPolicyMonitorControl monitorMaxDelay(4) is not enabled. When enabled the default value for this object is the max delay value specified in the associated action definition plus 10%. If the action definition doesn't have a max delay defined then there is no default for this object and a value MUST be specified prior to activating this entry when monitorMaxDelay(4) is selected."

::= { slapmPolicyMonitorEntry 13 }

slapmPolicyMonitorMaxDelayLow OBJECT-TYPE

SYNTAX Unsigned32
UNITS "milliseconds"
MAX-ACCESS read-create
STATUS current

DESCRIPTION

"The threshold for generating a slapmMaxDelayOkay notification, signalling that a monitored maximum delay rate has fallen to an acceptable level.

A value of zero for this object is returned when the slapmPolicyMonitorControl monitorMaxDelay(4) is not enabled. When enabled the default value for this object is the max delay value specified in the associated action definition minus 10%. If the action definition doesn't have a max delay defined then there is no default for this object and a value MUST be specified prior to activating this entry when monitorMaxDelay(4) is selected."

::= { slapmPolicyMonitorEntry 14 }

slapmPolicyMonitorMinRateNotAchieved OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

White, Kenneth

Expires December 1998

[Page 17]

"The number of times that a minimum transfer rate was not achieved."
 ::= { slapmPolicyMonitorEntry 15 }

slapmPolicyMonitorMaxRateExceeded OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times that a maximum transfer rate was exceeded."

::= { slapmPolicyMonitorEntry 16 }

slapmPolicyMonitorMaxDelayExceeded OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times that a maximum delay rate was exceeded."

::= { slapmPolicyMonitorEntry 17 }

slapmPolicyMonitorRowStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This object allows entries to be created and deleted in the slapmPolicyMonitorTable. An entry in this table is deleted by setting this object to destroy(6)."

::= { slapmPolicyMonitorEntry 18 }

-- TCP Connection Table Extensions

slapmTcpConnTable OBJECT-TYPE

SYNTAX SEQUENCE OF SlapmTcpConnEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Extends tcpConnTable to support SLA policies. This table doesn't augment the TCP connection table as defined in [RFC2012](#), since the order of the addresses has been changed and the table has been structured to support ipv6 addresses.

The indexing for this table is designed to support the use of an SNMP GET-NEXT operation using only

the remote address and remote port as a way for
a management station to retrieve the table entries
relating to a particular client."
::= { slapmObjects 6 }

slapmTcpConnEntry OBJECT-TYPE

SYNTAX SlapmTcpConnEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Describes a particular tcp connection entry. This table does not have slapmPolicyStatsOwnerIndex as part of its indexing since this table's contents is intended to span multiple users."

INDEX {

slapmTcpConnRemAddress,
slapmTcpConnRemPort,
slapmTcpConnLocalAddress,
slapmTcpConnLocalPort

}

::= { slapmTcpConnTable 1 }

SlapmTcpConnEntry ::=

SEQUENCE {

slapmTcpConnRemAddress	OCTET STRING,
slapmTcpConnRemPort	Unsigned32,
slapmTcpConnLocalAddress	OCTET STRING,
slapmTcpConnLocalPort	Unsigned32,
slapmTcpConnPolicyName	SlapmNameType,
slapmTcpConnTrafficProfileName	SlapmNameType,
slapmTcpConnLastActivity	TimeTicks,
slapmTcpConnInOctets	Counter32,
slapmTcpConnOutOctets	Counter32,
slapmTcpConnOutBufferedOctets	Counter32,
slapmTcpConnInBufferedOctets	Counter32,
slapmTcpConnReXmts	Counter32,
slapmTcpConnRoundTripTime	Unsigned32,
slapmTcpConnRoundTripVariance	Unsigned32,
slapmTcpConnInSegments	Counter32,
slapmTcpConnOutSegments	Counter32,
slapmTcpConnAppIName	SlapmNameType

}

slapmTcpConnRemAddress OBJECT-TYPE

SYNTAX OCTET STRING (SIZE(4 | 16))

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Indicate the remote address of a TCP connection.

A remote address can be either an ipv4 address in which case 4 octets are required or as an ipv6 address that requires 16 octets."

::= { slapmTcpConnEntry 1 }

slapmTcpConnRemPort OBJECT-TYPE
SYNTAX Unsigned32(0..65535)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION

White, Kenneth

Expires December 1998

[Page 19]

"Indicate the remote port of a TCP connection."
 ::= { slapmTcpConnEntry 2 }

slapmTcpConnLocalAddress OBJECT-TYPE

SYNTAX OCTET STRING (SIZE(4 | 16))

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Indicate the local address of a TCP connection.

A local address can be either an ipv4 address in which
case 4 octets are required or as an ipv6 address that
requires 16 octets."

::= { slapmTcpConnEntry 3 }

slapmTcpConnLocalPort OBJECT-TYPE

SYNTAX Unsigned32(0..65535)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Indicate the local port of a TCP connection."

::= { slapmTcpConnEntry 4 }

slapmTcpConnPolicyName OBJECT-TYPE

SYNTAX SlapmNameType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Policy name that this entry relates to."

::= { slapmTcpConnEntry 5 }

slapmTcpConnTrafficProfileName OBJECT-TYPE

SYNTAX SlapmNameType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The corresponding traffic profile name."

::= { slapmTcpConnEntry 6 }

slapmTcpConnLastActivity OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of 100ths of seconds since this entry
was last used."

DEFVAL { 0 }

::= { slapmTcpConnEntry 7 }

slapmTcpConnInOctets OBJECT-TYPE

SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"The number of bytes received from IP for this

White, Kenneth

Expires December 1998

[Page 20]


```
        connection."
 ::= { slapmTcpConnEntry 8 }

slapmTcpConnOutOctets OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The number of bytes sent to IP for this connection."
 ::= { slapmTcpConnEntry 9 }

slapmTcpConnOutBufferedOctets OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Number of outgoing bytes buffered"
 ::= { slapmTcpConnEntry 10 }

slapmTcpConnInBufferedOctets OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Number of incoming bytes buffered"
 ::= { slapmTcpConnEntry 11 }

slapmTcpConnReXmts OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Number of retransmissions"
 ::= { slapmTcpConnEntry 12 }

slapmTcpConnRoundTripTime OBJECT-TYPE
    SYNTAX      Unsigned32
    UNITS        "milliseconds"
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The amount of time that has elapsed, measured in
         milliseconds, from when the last TCP segment was
         transmitted by the TCP Stack until the ACK was
         received."
 ::= { slapmTcpConnEntry 13 }

slapmTcpConnRoundTripVariance OBJECT-TYPE
    SYNTAX      Unsigned32
```

MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Round trip time variance."
::= { slapmTcpConnEntry 14 }

White, Kenneth

Expires December 1998

[Page 21]

slapmTcpConnInSegments OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of TCP in segments for this connections."

::= { slapmTcpConnEntry 15 }

slapmTcpConnOutSegments OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of TCP out segments for this connections."

::= { slapmTcpConnEntry 16 }

slapmTcpConnAppIName OBJECT-TYPE

SYNTAX SlapmNameType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The application name associated with this entry if known,
otherwise a zero-length octet string is returned as the value
of this object."

::= { slapmTcpConnEntry 17 }

-- udpTable extension

slapmUdpTable OBJECT-TYPE

SYNTAX SEQUENCE OF SlapmUdpEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Extends udp table to support SLA policies. This
table doesn't use the AUGMENT option to define
an extension to the udp listen table as defined
in [RFC 2013](#) since the table has been structured to
support ipv6 addresses."

::= { slapmObjects 7 }

slapmUdpEntry OBJECT-TYPE

SYNTAX SlapmUdpEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Describes a particular udp entry."

INDEX {

```
        slapmUdpLocalAddress,  
        slapmUdpLocalPort  
    }  
    ::= { slapmUdpTable 1 }
```

White, Kenneth

Expires December 1998

[Page 22]

SlapmUdpEntry ::=

SEQUENCE

```
{
    slapmUdpLocalAddress      OCTET STRING,
    slapmUdpLocalPort         Unsigned32,
    slapmUdpPolicyName        SlapmNameType,
    slapmUdpTrafficProfileName SlapmNameType,
    slapmUdpLastAct           TimeTicks,
    slapmUdpInDgrams          Counter32,
    slapmUdpInOctets          Counter32,
    slapmUdpOutDgrams         Counter32,
    slapmUdpOutOctets         Counter32,
    slapmUdpAppName           SlapmNameType
}
```

slapmUdpLocalAddress OBJECT-TYPE

SYNTAX OCTET STRING (SIZE(4 | 16))

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Indicate the local address of a udp listener.

A local address can be either an ipv4 address in which case 4 octets are required or as an ipv6 address that requires 16 octets."

::= { slapmUdpEntry 1 }

slapmUdpLocalPort OBJECT-TYPE

SYNTAX Unsigned32(0..65535)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Indicate the local port of a udp listener."

::= { slapmUdpEntry 2 }

slapmUdpPolicyName OBJECT-TYPE

SYNTAX SlapmNameType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

""

::= { slapmUdpEntry 3 }

slapmUdpTrafficProfileName OBJECT-TYPE

SYNTAX SlapmNameType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

""

::= { slapmUdpEntry 4 }

slapmUdpLastAct OBJECT-TYPE
SYNTAX TimeTicks
UNITS "100s of seconds"
MAX-ACCESS read-only

White, Kenneth

Expires December 1998

[Page 23]

STATUS current

DESCRIPTION

"The amount of time that has elapsed since the corresponding entry has had any activity. Measured in hundreds of seconds."

::= { slapmUdpEntry 5 }

slapmUdpInDgrams OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of received datagrams."

::= { slapmUdpEntry 6 }

slapmUdpInOctets OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of octets received."

::= { slapmUdpEntry 7 }

slapmUdpOutDgrams OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of send datagrams."

::= { slapmUdpEntry 8 }

slapmUdpOutOctets OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of octets sent."

::= { slapmUdpEntry 9 }

slapmUdpAppName OBJECT-TYPE

SYNTAX SlapmNameType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The application name associated with this entry if known, otherwise a zero-length octet string is returned as the value of this object."

::= { slapmUdpEntry 10 }

-- Notifications

slapMinRateNotAchieved NOTIFICATION-TYPE
OBJECTS {

White, Kenneth

Expires December 1998

[Page 24]


```
    slapmPolicyMonitorIntTime,  
    slapmPolicyMonitorCurrentInRate,  
    slapmPolicyMonitorCurrentOutRate,  
    slapmPolicyMonitorMinRateNotAchieved,  
    slapmPolicyMonitorMaxRateExceeded,  
    slapmPolicyMonitorMaxDelayExceeded  
}  
STATUS    current  
DESCRIPTION  
    "This notification is generated when the minimum transfer  
    rate is not achieved with respect to a low threshold."  
 ::= { slapmNotifications 1 }
```

slapmMinRateOkay NOTIFICATION-TYPE

```
OBJECTS {  
    slapmPolicyMonitorIntTime,  
    slapmPolicyMonitorCurrentInRate,  
    slapmPolicyMonitorCurrentOutRate,  
    slapmPolicyMonitorMinRateNotAchieved,  
    slapmPolicyMonitorMaxRateExceeded,  
    slapmPolicyMonitorMaxDelayExceeded  
}  
STATUS    current  
DESCRIPTION  
    "This notification is generated when the minimum transfer  
    rate has increased to an acceptable level."  
 ::= { slapmNotifications 2 }
```

slapmMaxRateExceeded NOTIFICATION-TYPE

```
OBJECTS {  
    slapmPolicyMonitorIntTime,  
    slapmPolicyMonitorCurrentInRate,  
    slapmPolicyMonitorCurrentOutRate,  
    slapmPolicyMonitorMinRateNotAchieved,  
    slapmPolicyMonitorMaxRateExceeded,  
    slapmPolicyMonitorMaxDelayExceeded  
}  
STATUS    current  
DESCRIPTION  
    "This notification is generated when the maximum transfer  
    rate has been exceeded with respect to a high threshold."  
 ::= { slapmNotifications 3 }
```

slapmMaxRateOkay NOTIFICATION-TYPE

```
OBJECTS {  
    slapmPolicyMonitorIntTime,  
    slapmPolicyMonitorCurrentInRate,  
    slapmPolicyMonitorCurrentOutRate,  
    slapmPolicyMonitorMinRateNotAchieved,
```

```
        slapmPolicyMonitorMaxRateExceeded,  
        slapmPolicyMonitorMaxDelayExceeded  
    }  
    STATUS    current  
    DESCRIPTION
```

White, Kenneth

Expires December 1998

[Page 25]

"This notification is generated when the maximum transfer rate has fallen to an acceptable level."
 ::= { slapmNotifications 4 }

slapmMaxDelayExceeded NOTIFICATION-TYPE

OBJECTS {
 slapmPolicyMonitorIntTime,
 slapmPolicyMonitorCurrentInRate,
 slapmPolicyMonitorCurrentOutRate,
 slapmPolicyMonitorMinRateNotAchieved,
 slapmPolicyMonitorMaxRateExceeded,
 slapmPolicyMonitorMaxDelayExceeded
}
STATUS current
DESCRIPTION
 "This notification is generated when the maximum delay rate has been exceeded with respect to a high threshold."
 ::= { slapmNotifications 5 }

slapmMaxDelayOkay NOTIFICATION-TYPE

OBJECTS {
 slapmPolicyMonitorIntTime,
 slapmPolicyMonitorCurrentInRate,
 slapmPolicyMonitorCurrentOutRate,
 slapmPolicyMonitorMinRateNotAchieved,
 slapmPolicyMonitorMaxRateExceeded,
 slapmPolicyMonitorMaxDelayExceeded
}
STATUS current
DESCRIPTION
 "This notification is generated when the maximum delay rate has fallen to an acceptable level."
 ::= { slapmNotifications 6 }

slapmPolicyDeleteNeeded NOTIFICATION-TYPE

OBJECTS {
 slapmPolicyStatsOperStatus
}
STATUS current
DESCRIPTION
 "This notification is generated when a slapmPolicyStatsEntry via its slapmPolicyStatsOperStatus object has transitioned into the deleted(3) state and can't be deleted because other tables have dependent entries."
 ::= { slapmNotifications 7 }

-- Conformance information

-- Compliance statements

slapmCompliances OBJECT IDENTIFIER ::= { slapmConformance 1 }

slapmGroups OBJECT IDENTIFIER ::= { slapmConformance 2 }

White, Kenneth

Expires December 1998

[Page 26]

```
-----  
-- Compliance statements  
-----
```

```
slapmCompliance MODULE-COMPLIANCE
```

```
    STATUS current
```

```
    DESCRIPTION
```

```
        "The compliance statement for the SLAPM-MIB."
```

```
    MODULE -- this module
```

```
        MANDATORY-GROUPS {
```

```
            slapmBaseGroup,  
            slapmNotGroup
```

```
        }
```

```
        GROUP slapmEndSystemGroup
```

```
        DESCRIPTION
```

```
            "The contents of this group is required by end-system  
            implementations."
```

```
 ::= { slapmCompliances 1 }
```

```
-----  
-- MIB groupings  
-----
```

```
slapmBaseGroup OBJECT-GROUP
```

```
    OBJECTS {
```

```
        slapmSpinLock,  
        slapmPolicyQueryCount,  
        slapmPolicyAccessCount,  
        slapmPolicySuccessAccessCount,  
        slapmPolicyNotFoundCount,  
        slapmPolicyPurgeTime,  
        slapmPolicyStatsOperStatus,  
        slapmPolicyStatsActiveConns,  
        slapmPolicyStatsFirstActivated,  
        slapmPolicyStatsLastMapping,  
        slapmPolicyStatsLastActivity,  
        slapmPolicyStatsInOctets,  
        slapmPolicyStatsOutOctets,  
        slapmPolicyStatsConnectionLimit,  
        slapmPolicyStatsActiveConnections,  
        slapmPolicyStatsAcceptCount,  
        slapmPolicyStatsDenyCount,  
        slapmPolicyMonitorControl,  
        slapmPolicyMonitorInterval,  
        slapmPolicyMonitorIntTime,  
        slapmPolicyMonitorCurrentInRate,  
        slapmPolicyMonitorCurrentOutRate,  
        slapmPolicyMonitorMinRateLow,
```

slapmPolicyMonitorMinRateHigh,
slapmPolicyMonitorMaxRateHigh,
slapmPolicyMonitorMaxRateLow,
slapmPolicyMonitorMaxDelayHigh,
slapmPolicyMonitorMaxDelayLow,

White, Kenneth

Expires December 1998

[Page 27]

```
    slapmPolicyMonitorMinRateNotAchieved,  
    slapmPolicyMonitorMaxRateExceeded,  
    slapmPolicyMonitorMaxDelayExceeded,  
    slapmPolicyMonitorRowStatus  
}
```

STATUS current

DESCRIPTION

"The group of objects defined by this MIB that are
required for all implementations to be compliant."

::= { slapmGroups 1 }

slapmEndSystemGroup OBJECT-GROUP

OBJECTS {

```
    slapmTcpConnPolicyName,  
    slapmTcpConnTrafficProfileName,  
    slapmTcpConnLastActivity,  
    slapmTcpConnInOctets,  
    slapmTcpConnOutOctets,  
    slapmTcpConnOutBufferedOctets,  
    slapmTcpConnInBufferedOctets,  
    slapmTcpConnReXmts,  
    slapmTcpConnRoundTripTime,  
    slapmTcpConnRoundTripVariance,  
    slapmTcpConnInSegments,  
    slapmTcpConnOutSegments,  
    slapmTcpConnApplName,  
    slapmUdpPolicyName,  
    slapmUdpTrafficProfileName,  
    slapmUdpLastAct,  
    slapmUdpInDgrams,  
    slapmUdpInOctets,  
    slapmUdpOutDgrams,  
    slapmUdpOutOctets,  
    slapmUdpApplName
```

}

STATUS current

DESCRIPTION

"The group of objects defined by this MIB that are
required for end system implementations."

::= { slapmGroups 2 }

slapmNotGroup NOTIFICATION-GROUP

NOTIFICATIONS {

```
    slapmMinRateNotAchieved,  
    slapmMinRateOkay,  
    slapmMaxRateExceeded,  
    slapmMaxRateOkay,  
    slapmMaxDelayExceeded,  
    slapmMaxDelayOkay,
```

```
        slapmPolicyDeleteNeeded
    }
STATUS    current
DESCRIPTION
```

"The group of notifications defined by this MIB that are

White, Kenneth

Expires December 1998

[Page 28]


```
    required for all implementations to be compliant."  
 ::= { slapmGroups 3 }
```

END

5.0 Security Considerations

Certain management information defined in this MIB may be considered sensitive in some network environments. Therefore, authentication of received SNMP requests and controlled access to management information SHOULD be employed in such environments. The method for this authentication is a function of the SNMP Administrative Framework, and has not been expanded by this MIB.

It is RECOMMENDED that creation of entries in the slapmPolicyMonitorTable not be allowed in insecure environments.

6.0 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 implementers 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.

7.0 Acknowledgments

This document is a product of the IETF.

8.0 References

- [1] Case, J., M. Fedor, M. Schoffstall, J. Davin, "Simple Network Management Protocol", [RFC 1157](#), SNMP Research, Performance Systems International, MIT Laboratory for Computer Science, May 1990.

- [2] McCloghrie, K., and M. Rose, Editors, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II", STD 17, [RFC 1213](#), Hughes LAN Systems, Performance Systems International, March 1991.
- [3] Case, J., McCloghrie, K., Rose, M., and Waldbusser S., "Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)", [RFC 1902](#), January 1996.
- [4] Case, J., McCloghrie, K., Rose, M., and Waldbusser, S., "Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)", [RFC 1903](#), January 1996.
- [5] Case, J., McCloghrie, K., Rose, M., and Waldbusser, S., "Conformance Statements for Version 2 of the Simple Network Management Protocol (SNMPv2)", [RFC 1904](#), January 1996.
- [6] Case, J., McCloghrie, K., Rose, M., and Waldbusser, S., "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)", [RFC 1905](#), January 1996.
- [7] Harrington D., Presuhn, R., Wijnen, B., "An Architecture for Describing SNMP Management Frameworks", [RFC 2271](#), Cabletron Systems, BMC Software, Inc., IBM T.J. Watson Research, January 1998.
- [8] Harrington D., Presuhn, R., Wijnen, B., "Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)", [RFC 2272](#), Cabletron Systems, BMC Software, Inc., IBM T.J. Watson Research, January 1998.
- [9] Levi D., Meyer P., Stewart, B., "SNMPv3 Applications", [RFC 2273](#), SNMP Research, Inc., Secure Computing Corporation, Cisco Systems, January 1998.
- [10] Blumenthal, U., Wijnen, B., "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", [RFC 2274](#), IBM T. J. Watson Research, January 1998.
- [11] Wijnen, B., Presuhn, R., McCloghrie, K., "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)", [RFC 2275](#), IBM T.J. Watson Research, BMC Software, Inc., Cisco Systems, Inc., January 1998.
- [12] Hovey, R., and S. Bradner, "The Organizations Involved in the IETF Standards Process", [BCP 11](#), [RFC 2028](#), October 1996.
- [13] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

- [14] 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.

- [15] Rose, M., and K. McCloghrie, "Concise MIB Definitions", [RFC 1212](#), Performance Systems International, Hughes LAN Systems, March 1991.
- [16] M. Rose, "A Convention for Defining Traps for use with the SNMP", [RFC 1215](#), Performance Systems International, March 1991.
- [17] 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.
- [18] 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.
- [19] "Schema for Service Level Administration of Differentiated Services and Integrated Services in Networks", [<draft-ellesson-sla-schema-02.txt>](#), June 1, 1998.

[9.0](#) Author's Address

Kenneth D. White
Dept. BRQA/Bldg. 501/G114
IBM Corporation
P.O.Box 12195
3039 Cornwallis
Research Triangle Park, NC 27709, USA
E-mail: kennethw@vnet.ibm.com

[10.0](#) Full Copyright Statement

Copyright (C) The Internet Society (1997). 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.

White, Kenneth

Expires December 1998

[Page 31]

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.

White, Kenneth

Expires December 1998

[Page 32]