SNMP managed objects for Middlebox Communications (MIDCOM)

<draft-srisuresh-midcom-mib-00.txt>

Abstract

Middlebox communication (midcom) was conceived to move application level gateway (ALG) intelligence out of middleboxes into application specific midcom agents. Midcom agents will be assumed to use midcom to control middlebox resources so as to permit applications to traverse a middlebox. The scope of the middleboxes is limited to NAT and firewall devices. This document defines SNMP managed midcom objects to control middlebox resources and justifies adapting SNMPv3 as the midcom protocol.
Table of Contents

1. Overview.................................................................2
2. Terminology...............................................................3
2.1. "Midcom agent" or "agent"...........................................3
2.2. SNMP agent.............................................................3
2.3. NAT session...........................................................3
3. SNMP Management Framework..........................................4
4. MIDCOM Overview and SNMP Applicability..........................4
5. SNMP and the MIDCOM data model.....................................5
   5.1 Secure Communications...........................................7
   5.2 Device Configuration..............................................8
   5.3 Service Configuration............................................8
   5.4 Midcom compatibility requirements on NAT and Firewall.......9
6. Midcom MIB....................................................................10
7. Security Considerations..................................................45
8. Acknowledgements...........................................................45
9. References.......................................................................45
   Normative References....................................................45
   Informative References..................................................47
Author’s address...............................................................48
Full Copyright Statement....................................................48

1. Overview

The principal objective of the document is to describe how SNMPv3 may be adapted as the MIDCOM protocol. MIDCOM MIB is defined to facilitate transactions between a midcom agent and a middlebox.

The scope of the middleboxes considered in the document is limited to NAT and Firewall devices. This document refers external documents for NAT and firewall MIBs and states the compliance criteria for the external MIBS to be MIDCOM compliant.

Section 1 provides an overview of the SNMP Management Framework.
Section 2 provides further background on SNMP and its applicability to the MIDCOM Protocol Framework, Requirements and semantics.

Section 3 provides a high level overview of the SNMPv3 protocol, the MIB data model and its applicability together as a MIDCOM protocol.

Section 6 has the midcom mib described in detail.

2. Terminology
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].

The Midcom terms used throughout this document are mostly as per RFC 3303. The NAT terms used in the document are mostly as per RFC 2663. Definition for the term "Symmetric NAT" may be found in RFC 3489. Symmetric NAT is a variation of NAPT in that a port bind is not retained across multiple sessions from the same private source port. The following terms used extensively in the document are reiterated here for clarity.

2.1. "Midcom agent" or "agent"

Midcom agent, hereafter referred simply as agent, is an entity performing ALG functions, logically external to a middlebox. MIDCOM agents possess a combination of application awareness and knowledge of the middlebox function.

A midcom agent may be located anywhere in the end-2-end path of an application path, including the middlebox itself. The exact interface through which a midcom agent engages in a midcom session with the middlebox is irrelevant to the enforcement of midcom.

2.2. SNMP agent

SNMP agent is an entity on middlebox servicing SNMP requests from SNMP applications, including midcom agents.

2.3. NAT session

A NAT session is an association between a session as seen in the private realm and a session as seen in the public realm, by virtue of NAT translation. If a session in the private realm were to be represented as (PrivateSrcAddr, PrivateDstAddr, TransportProtocol, PrivateSrcPort, PrivateDstPort) and the same session in the public realm were to be represented as (PublicSrcAddr, PublicDstAddr, TransportProtocol, PublicSrcPort, PublicDstPort), the NAT session will provide the translation glue between the two session representations.

3. SNMP Management Framework

For a detailed overview of the documents that describe the current Internet-Standard (SNMP) Management Framework, please refer to...
section 7 of RFC 3410 [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIv2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

4. MIDCOM Overview and SNMP Applicability

The MIDCOM architecture and framework [RFC3303] defines a model in which trusted third parties can be delegated to assist middleboxes in performing their operations, without requiring application intelligence be embedded in the middleboxes. This trusted third party is referred to as the MIDCOM Agent. The MIDCOM protocol is defined between the MIDCOM agent and middlebox.

The SNMP management framework provides functions equivalent to those defined by the MIDCOM framework, although there are a few architectural differences.

For SNMP, application intelligence is captured in MIB modules, rather than in the messaging protocol. MIB modules define a data model of the information that can be collected and configured for managed functionality. The SNMP messaging protocol transports the data in a standardized format without needing to understand the semantics of the data being transferred. The endpoints of the communication understand the semantics of the data.

Traditionally, the SNMP endpoints have been called Manager and Agent. An SNMP manager is an entity capable of generating requests and receiving notifications, and a SNMP agent is an entity capable of responding to requests and generating notifications. As applied to the MIDCOM framework, the SNMP Manager corresponds to the MIDCOM agent and the SNMP Agent corresponds to the Middlebox.

The MIDCOM protocol is divided into three phases, per section 4 of [RFC3303]:

- Session Setup
- Run-time (involving real-time configuration of the middlebox)
- Session Termination

A MIDCOM session is defined to be a lasting association between a MIDCOM agent and a middlebox. The MIDCOM agent should initiate
the session prior to the start of the application. Although the SNMP management framework does not have the concept of a session, session-like associations can be established through the use of managed objects. Requests from the MIDCOM agent to the Middlebox are performed using Read/write access to managed objects defined in MIB modules. The middlebox (SNMP agent) responds to requests by sending an SNMP response message indicating the success or failure of the request. The MIDCOM agent (SNMP manager) MAY verify this information by reading or polling the corresponding managed objects.

The MIDCOM Protocol semantics [MDCSEM] defines two basic transaction types: request transactions and notify transactions. SNMPv3 uses the architecture detailed in [RFC3411], where all SNMP entities are capable of performing certain functions, such as the generation of requests, response to requests, the generation of asynchronous notifications and the receipt of notifications. SNMP is used to read and manipulate a virtual database (the MIB) which is composed of objects representing commands, controls, status, and statistics, which are defined in managed-application-specific MIB modules.

5. SNMPv3 for use as MIDCOM protocol

The following diagram (Figure 1) is an operational model assumed by the MIDCOM protocol. Requirements on the Midcom protocol is identified by the MIDCOM protocol framework, requirements and semantics documents. Specification of policies via the MIDCOM PDP is outside the scope of the MIDCOM protocol and is omitted in the discussion in the remainder of this document.
Legend: .... Middlebox Communication Protocol (MIDCOM)
//// MIDCOM PDP Interface (outside scope of this document)
**** Managed objects relevant to the MIDCOM Interface
(with the associated letters referencing the MIB modules potentially applicable summarized below:

Figure 1: operational model assumed by the MIDCOM protocol
5.1 SNMP MIB data model on a middlebox

The following diagram (Figure 2) restates the Midcom operational model when SNMPv3 is adapted as the Midcom protocol. The SNMP based model below includes midcom MIB and middlebox function MIBs objects. These MIBs are described in detail in the remainder of this document.
Legend: .... SNMP used as the MIDCOM protocol
---- Interface between the SNMP agent and
the MIB modules.
**** The MIB methods of the Midcom MIB
accessing middlebox function specific
objects.

Figure 2: SNMPv3 operating as the Midcom protocol

5.2 Secure Communications

MIDCOM requirements include mutual authentication, message integrity checking, timeliness checking to prevent replay, message encryption, and authorization controls to ensure only certain agents can modify certain subsets of middlebox configurations. MIDCOM requires secure request-response capabilities and secure notifications.

SNMPv3 is designed to provide secure communications between two end-points. SNMPv3 defines MIB modules to allow the monitoring and configuration of all these security features. They are defined in RFC3411-RFC3418, and RFC3410 provides an overview of these capabilities.

5.3. Midcom functions

Midcom MIB does not assume a middlebox to have implemented MIBs (standard or vendor proprietary) for NAT and firewall functions. Middlebox functions may be configured and managed independently of the midcom MIB. However, midcom MIB will have rule-change parameters and a pointer to the FW/NAT MIB objects (even if vendor proprietary). The FW and NAT MIBS actually contain the detailed objects. For instance, multiple agents might end up using the same NAT BIND, yet each agent might define their own Lifetime parameter and directionality for the bind. As a result, the agent specific Bind identifier is set uniquely, independent of the NAT native bind. Yet, the agent specific bind has a pointer to the NAT bind.

Midcom MIB below is designed to meet the midcom requirements (RFC 3304). A set of MIB objects, one per each middlebox resource type, are defined to run midcom transactions. The resulting resources, along with rule-changing parameters and a pointer to FW/NAT MIB objects are maintained as MIB tables, one for each resource type. Also defined are group based transaction objects and group tables, as required by RFC
5.3.1. Agent registration for notification

midcomAgentTable is designed to include all the agents that engage in a midcom session with the middlebox. Each active row of the table corresponds to a midcom agent. The agent includes the notify parameters within this row to allow middleboxes to send asynchronous notifications back to the agent. Also included is an agent-unique Middlebox Identifier a middlebox should use to identify itself during the notifications.

5.3.2. Middlebox Configuration for midcom

Not every middlebox is required to enable midcom on all its interfaces. midcomConfig is designed to configure midcom on a per-interface basis on a middlebox.

5.3.3. Midcom transactions and relevant tables

Midcom transactions may be divided into group transactions and resource transactions. A transaction is atomic and the results of a transaction are saved into relevant tables at the end of the transaction. Results of a transaction conducted by an agent may be reviewed anytime prior to executing another transaction of the same kind by the same agent.

midcomTransGroupTable is defined to allow multiple agents to simultaneously add or delete Group identifiers and set group-wide parameters such as LifeTime and MaxIdletime. Results of the transaction are transferred into midcomGroupTable for later reference and further parameter modification by the agent.

midcomTransBindTable, midcomTransNatSessionTable, and midcomTransFilterTable are defined to allow multiple agents to simultaneously request middlebox resources and set parameters such as LifeTime and MaxIdletime. Results of the transactions are transferred respectively into the relevant resource table, namely midcomBindTable, midcomNatSessionTable and midcomFilterTable for later reference and further parameter modification by the agent.

5.4. Midcom compatibility requirements on NAT and Firewall

Middlebox function resources (bind, NatSession and firewall
filter) are now required to carry an additional LifeTime parameter.

Given that there may be several agents referring the same resource (ex: bind) and each agent may choose to control lifetime, MaxIdleTime and Bind orientation as appropriate for the agent, the middlebox function is now required to use a superset of the settings. Further, a new AgentCount will be required to track the number of agents referring a certain resource.

As for notification, middlebox functions might retain a pointer to the first active agent and the active agents referring the same resource might link between themselves. Doing this will ensure that Midcom is able to send notifications to all affected agents when required to do by the middlebox function.

Agent precedence and inter-agent overlap on the use of resources could be particularly tricky in the case of firewall rules. For example, essentially the same filter can be configured by multiple agents with different priorities (assume, highest or lowest is all that a midcom transaction will specify). The last rule will take precedence, potentially overruling the previous agent transactions. Further, when some of the filters are specific and some are more general, there can be undesired ordering of the filters. Agents are advised to include specific rules, so as not to overrule or be overridden by other filter rules.

6.0. Midcom MIB

Midcom MIB provides a means for midcom agents to control middlebox resources and for middlebox to asynchronously notify the midcom agents of relevant state changes. Midcom agents learn of the functions present on the middlebox using this MIB.

midcom-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY,
    OBJECT-TYPE,
    NOTIFICATION-TYPE,
    Integer32,
    Unsigned32,
    Gauge32,
Counter64, TimeTicks, mib-2
FROM SNMPv2-SMI -- RFC 2578

TEXTUAL-CONVENTION,
StorageType, RowStatus, TimeInterval
FROM SNMPv2-TC -- RFC 2579

MODULE-COMPLIANCE,
NOTIFICATION-GROUP,
OBJECT-GROUP
FROM SNMPv2-CONF -- RFC 2580

ifIndex, InterfaceIndex
FROM IF-MIB -- RFC 2863

SnmpAdminString
FROM SNMP-FRAMEWORK-MIB -- RFC 3411

InetAddressType, InetAddress, InetPortNumber
FROM INET-ADDRESS-MIB; -- RFC 3291

NatTranslationEntity, NatBindIdOrZero, NatSessionId,
FROM NAT-MIB;

midcomMIB MODULE-IDENTITY
LAST-UPDATED "200310200000Z"
ORGANIZATION "IETF Midcom Working Group"
CONTACT-INFO
"WG charter:
http://www.ietf.org/html.charters/midcom-charter.html

Mailing Lists:
  General Discussion: midcom@ietf.org
  To Subscribe: midcom-request@ietf.org
  In Body: subscribe your_email_address

Author:
  Pyda Srisuresh
  1179-A North McDowell Blvd.
Petaluma, CA 94954
DESCRIPTION
"This MIB module defines the managed objects for midcom."

REVISION     "200310200000Z" -- 20th Sept. 2003
DESCRIPTION
"Initial version of this MIB module."
::= { mib-2 XXX } -- RFC Ed.: replace XXX with IANA-assigned
   -- number & remove this note

midcomMIBObjects OBJECT IDENTIFIER ::= { midcomMIB 1 }

--
-- Four Groups
--
-- o midcomConfig       - Configuration of a middlebox for
--                        midcom access.
-- o midcomAgentInfo    - Active agent info, including the info
--                        necessary for asynchronous notification.
-- o midcomTables       - Results of agent initiated transactions
--                        are saved into relevant tables for later
--                        reference and parameter modification by
--                        the agents.
-- o midcomTransactions - Midcom agent initiated transactions.
--

midcomConfig OBJECT IDENTIFIER ::= 
   { midcomMIBObjects 1 }

midcomAgentInfo OBJECT IDENTIFIER ::= 
   { midcomMIBObjects 2 }

midcomTables OBJECT IDENTIFIER ::= 
   { midcomMIBObjects 3 }

midcomTransactions OBJECT IDENTIFIER ::= 
   { midcomMIBObjects 4 }

--
-- Textual conventions used
--
MidcomMBFunctionEnum ::= TEXTUAL-CONVENTION
STATUS       current
DESCRIPTION
"An enumeration of Middlebox functions that are
supported by the midcom protocol. Inclusion of values is not intended to imply that those functions need to be supported. Any change in this TEXTUAL-CONVENTION should also be reflected in the definition of midcomConfMBFunctionType object which is a BITS representation of this TEXTUAL-CONVENTION."

SYNTAX INTEGER {
    none (1), -- not specified
    nat (2),
    firewall (3)
}

MidcomMBFunctionBITS ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
"A BITS representation of Middlebox functions for which MIDCOM is enabled on a middlebox. Any change in this TEXTUAL-CONVENTION should also be reflected in the definition of midcomConfMBFunctionEnum object which is an enumeration of the middlebox functions supported" 

SYNTAX BITS {
    nat (0),
    firewall (1)
}

MidcomMBResource ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
"An enumeration of Middlebox function specific resource types that are supported by the midcom protocol. Inclusion of values is not intended to imply that those functions need to be supported."

SYNTAX INTEGER {
    none (1), -- not specified
    natBind(2),
    natSession(3),
    firewallFilter(4)
}

MidcomAgentIndex ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
"A unique id that is assigned to each midcom
session by the middlebox."

SYNTAX   Unsigned32 (1..4294967295)

MidcomBindMode ::= TEXTUAL-CONVENTION
STATUS       current
DESCRIPTION  "An indication of whether a bind is address bind
or port bind.
"
SYNTAX   INTEGER {
         addressBind (1),
         portBind    (2)
    }

midcomConf
-- The Configuration Group
-- The per-interface Midcom Configuration Table

midcomConfInterfaceTable OBJECT-TYPE
SYNTAX      SEQUENCE OF MidcomConfInterfaceEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION  "This table specifies the midcom configuration
attributes per interface on a device supporting
midcom access."
::= { midcomConfig 1 }

MidcomConfInterfaceEntry OBJECT-TYPE
SYNTAX      MidcomConfInterfaceEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION  "Each entry in the midcomConfInterfaceTable
holds a set of Midcom configuration parameters
pertaining to an interface"
INDEX   { ifIndex }
::= { midcomConfInterfaceTable 1 }

MidcomConfInterfaceEntry ::= SEQUENCE {
    midcomConfMBFunctionType       MidcomMBFunctionBITS,
    midcomConfStorageType          StorageType,
    midcomConfRowStatus            RowStatus
}
midcomConfMBFunctionType OBJECT-TYPE
SYNTAX      MidcomMBFunctionBits
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  "Middlebox functions for which Midcom processing is enabled."
::= { midcomConfInterfaceEntry 1 }

midcomConfStorageType OBJECT-TYPE
SYNTAX      StorageType
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  "The storage type for this conceptual row."
REFERENCE    "Textual Conventions for SMIv2, Section 2."
DEFVAL { nonVolatile }
::= { midcomConfInterfaceEntry 2 }

midcomConfRowStatus OBJECT-TYPE
SYNTAX      RowStatus
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  "The status of this conceptual row. None of the objects in this row may be modified while the value of this object is active(1)."
REFERENCE    "Textual Conventions for SMIv2, Section 2."
::= { midcomConfInterfaceEntry 3 }

--
--
-- midcomAgentInfo
--   Agent specific tables managed by the midcom MIB.
--
--
midcomAgentIndexNext OBJECT-TYPE
SYNTAX      MidcomAgentIndex
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  "When retrieved, this object returns an unused index into Agent table for the USM user that issued the read-request. The returned value can be used for creating a new entry in the Agent table."
::= { midcomAgentInfo 1 }
in the midcomAgentTable. The same return value also serves to create new entries in midcomTransGroup, midcomTransBind, midcomTransSession & midcomTransFilter tables. In all these tables, the first index would be set to the AgentIndex returned here and is set to read-only.

A value returned when reading this object is not returned again on subsequent read-requests as long as possible. This ensures that the same USM user can engage in multiple independent midcom sessions with the middlebox. Each midcom agent might be responsible for a different application.

::=  { midcomAgentInfo 1 }

--
-- midcomAgentTable
--  Agent Registration with Middlebox with
--  all the requisite information for notification.
--

midcomAgentTable OBJECT-TYPE
SYNTAX       SEQUENCE OF MidcomAgentEntry
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "Lists the active Midcom agents."
::=  { midcomAgentInfo 2 }

midcomAgentEntry OBJECT-TYPE
SYNTAX      MidcomAgentEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION  "Each entry in the midcomAgentTable pertains to a midcom agent. Parameters associated with the midcom agent are stored in this table. Each entry contains objects describing where notifications are to be sent to the MIDCOM agent."

INDEX   { midcomAgentIndex }
::=  { midcomAgentTable 1 }

MidcomAgentEntry ::= SEQUENCE {
  midcomAgentIndex MidcomAgentIndex,
  midcomAgentName MidcomNameOrPassword,
  midcomAgentMBId Unsigned32,
  midcomAgentAddrType InetAddressType,
  midcomAgentAddress InetAddress,

Srisuresh [Page 16]
midcomAgentIndex OBJECT-TYPE
SYNTAX MidcomAgentIndex
MAX-ACCESS read-only
STATUS current
DESCRIPTION "A middlebox-unique index or Identifier for each midcom agent in the Table. This object allows the same USM user to engage in multiple midcom sessions, perhaps one for each application. Each midcom agent will have a unique agentIndex."
::= { midcomAgentEntry 1 }

midcomAgentName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE (1..32))
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "The name of the SNMP manager that represents the midcom agent in this midcomAgentTable."
::= { midcomAgentEntry 2 }

midcomAgentMBId OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "This is a agent-unique Identifier issued by agent to the middlebox.

This identifier is to be used by the middlebox during asynchronous notifications to the agent."
::= { midcomAgentEntry 3 }

midcomAgentAddrType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "This object specifies the address type used for midcomAgentEntryAddress"
::= { midcomAgentEntry 4 }
midcomAgentAddress OBJECT-TYPE
SYNTAX InetAddress (SIZE (0..20))
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "This object represents the network layer
   address of the Midcom agent. This address, in
   conjunction with AddrType and the UDP port
   midcomAgentPort may be used by the middlebox
   functions for asynchronous notification to the
   agent.
"
 ::= { midcomAgentEntry 5 }

midcomAgentPort OBJECT-TYPE
SYNTAX InetPortNumber,
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "This object represents the UDP port of the
   Midcom agent. The combinations of (AddressType,
   Address, Port) are to be used by the middlebox
   functions for asynchronous notification to the
   agent.
"
 ::= { midcomAgentEntry 6 }

midcomAgentStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION "The status of this conceptual row.
   Objects in this row may be modified
   while the value of this object is active(1)."
REFERENCE "Textual Conventions for SMIv2, Section 2
 ::= { midcomAgentEntry 7 }

--
-- midcomTables   - Results of agent initiated transactions
-- are saved into relevant tables for later
-- reference and parameter modification by
-- the agents.
--
--
-- midcomGroupTable
group Ids per each agent.

midcomGroupTable OBJECT-TYPE
SYNTAX       SEQUENCE OF MidcomGroupEntry
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "Lists the groups registered by each agent."
::=    { midcomTables 1 }

midcomGroupEntry OBJECT-TYPE
SYNTAX       MidcomGroupEntry
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "Each entry in the GroupTable holds a unique tuple of parameters associated with a group Identifier. Group identifiers are registered by an agent with midcom."
INDEX   { midcomGroupAgentIndex,
midcomGroupMBResource,    
midcomGroupId } 
::= { midcomGroupTable 1 }

MidcomGroupEntry ::= SEQUENCE { 
    midcomGroupAgentIndex  MidcomAgentIndex,    
    midcomGroupMBResource  MidcomMBResource,    
    midcomGroupId     Unsigned32,    
    midcomGroupLifetime  TimeInterval,    
    midcomGroupMaxIdletime TimeInterval,    
    midcomGroupStatus    RowStatus    }

midcomGroupAgentIndex OBJECT-TYPE
SYNTAX       MidcomAgentIndex
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "Unique Identifier for an agent in the table"
::= { midcomGroupEntry 1 }

midcomGroupMBResource OBJECT-TYPE
SYNTAX       MidcomMBResource
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "Middlebox resource type for which the GroupId is registered by the agent."
::= { midcomGroupEntry 2 }
midcomGroupGroupId OBJECT-TYPE
SYNTAX       Unsigned32
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "A unique Group Identifier registered by the
agent for the resource the agent owns."
 ::= { midcomGroupEntry 3 }

midcomGroupLifetime OBJECT-TYPE
SYNTAX       TimeInterval
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  "Default Lifetime of the resources that are
assigned this group Id."
 ::= { midcomGroupEntry 4 }

midcomGroupMaxIdletime OBJECT-TYPE
SYNTAX       TimeInterval
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  "Default MaxIdletime of the resources that
are assigned this group Id."
 ::= { midcomGroupEntry 5 }

midcomGroupStatus  OBJECT-TYPE
SYNTAX      RowStatus
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  "The status of this conceptual row.
Objects in this row may be modified
while the value of this object is active(1)."
REFERENCE  "Textual Conventions for SMIV2, Section 2"
 ::= { midcomGroupEntry 6 }

-- midcomBindTable
--      Bind Ids managed by each agent.
--
midcomBindTable OBJECT-TYPE
SYNTAX       SEQUENCE OF MidcomBindEntry
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "Lists NAT binds owned by each agent."
midcomBindEntry OBJECT-TYPE
SYNTAX     MidcomBindEntry
MAX-ACCESS read-write
STATUS     current
DESCRIPTION "Each entry in the BindTable holds a unique tuple of parameters associated with a Bind."
INDEX     { midcomBindAgentIndex,
             midcomBindGroupId,
             midcomBindId }
 ::= { midcomBindTable 1 }

MidcomBindEntry ::= SEQUENCE {
    midcomBindAgentIndex           MidcomAgentIndex,
    midcomBindGroupId              Unsigned32,
    midcomBindId                   NatBindId,
    midcomBindLifetime             TimeInterval,
    midcomBindMaxIdleTime          TimeInterval,
    midcomBindIfIndex              InterfaceIndex,
    midcomBindTranslationEntity    NatTranslationEntity,
    midcomBindMBId                 NatBindId,
    midcomBindMode                 MidcomBindMode,
    midcomBindStatus               RowStatus
}

midcomBindAgentIndex OBJECT-TYPE
SYNTAX     MidcomAgentIndex
MAX-ACCESS read-only
STATUS     current
DESCRIPTION "Unique Identifier for an agent in the table"
 ::= { midcomBindEntry 1 }

midcomBindGroupId OBJECT-TYPE
SYNTAX     Unsigned32
MAX-ACCESS read-create
STATUS     current
DESCRIPTION "Group Identifier assignd to this bind resource.

A value of 0 implies that the bind does not belong to a group membership."
 ::= { midcomBindEntry 2 }

::= { midcomTables 2 }
midcomBindId OBJECT-TYPE
SYNTAX        NatBindId
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION   "Unique Bind Identifier assigend to this midcom
bind resource. This identifier is independent
of the bind identifier midcomBindMBId that is
managed by the NAT middlebox.
"
 ::= { midcomBindEntry 3 }

midcomBindLifetime OBJECT-TYPE
SYNTAX        TimeInterval
MAX-ACCESS    read-create
STATUS        current
DESCRIPTION   "Lifetime of the bind resource. When this is set to 0 and GroupId
is set to non-zero, the Lifetime of the
GroupId is used to determine the
lifetime of this resource.
"
 ::= { midcomBindEntry 4 }

midcomBindMaxIdletime OBJECT-TYPE
SYNTAX        TimeInterval
MAX-ACCESS    read-create
STATUS        current
DESCRIPTION   "MaxIdletime of the Bind resource. When this is set to 0 and GroupId
is set to non-zero, the MaxIdletime of the
GroupId is used to determine the
Maxidletime of this resource.
"
 ::= { midcomBindEntry 5 }

midcomBindIfIndex OBJECT-TYPE
SYNTAX        InterfaceIndex
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION   "Interface Index for which the bind is defined.
This value may be set to 0 to mean any
IP interface on the middlebox. This value
may also be set to 0, when the middlebox has
just one interface on which midcom is
configured.
"
 ::= { midcomBindEntry 6 }
midcomBindTranslationEntity OBJECT-TYPE
SYNTAX     NatTranslationEntity
MAX-ACCESS read-create
STATUS     current
DESCRIPTION "This object represents the direction of the session for which this BIND is applicable and entity within the first packet that is subject to translation."
::= { midcomBindEntry 7 }

midcomBindMBId OBJECT-TYPE
SYNTAX       NatBindId
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "Unique Bind Identifier managed by the NAT middlebox function. This identifier is independent of the bind identifier midcomBindId that is used in conjunction with midcom. Multiple midcomBindIds may be associated with the same midcomBindMBId."
::= { midcomBindEntry 8 }

midcomBindMode  OBJECT-TYPE
SYNTAX       MidcomBindMode
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "Indicates whether the bind is address bind or port bind."
::= { midcomBindEntry 9 }

midcomBindStatus  OBJECT-TYPE
SYNTAX      RowStatus
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"The status of this conceptual row. Objects in this row may be modified while the value of this object is active(1)."
REFERENCE
"Textual Conventions for SMIv2, Section 2"
::= { midcomBindEntry 6 }

--
-- midcomNatSessionTable
-- NAT Session Ids per each agent.

midcomNatSessionTable OBJECT-TYPE
SYNTAX      SEQUENCE OF MidcomNatSessionEntry
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION "Lists NAT sessions owned by each agent."
::=    { midcomTables 3 }

midcomNatSessionEntry OBJECT-TYPE
SYNTAX      MidcomNatSessionEntry
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION "Each entry in the NatSessionTable holds a
unique tuple of parameters associated with
a NAT session."

INDEX   { midcomNatSessionAgentIndex,
midcomNatSessionGroupId,
midcomNatSessionId }
::= { midcomNatSessionTable 1 }

MidcomNatSessionEntry ::= SEQUENCE {
    midcomNatSessionAgentIndex           MidcomAgentIndex,
    midcomNatSessionGroupId              Unsigned32,
    midcomNatSessionId                   NatSessionId,
    midcomNatSessionLifetime             TimeInterval,
    midcomNatSessionMaxIdleTime          TimeInterval,
    midcomNatSessionIfIndex              InterfaceIndex,
    midcomNatSessionRowStatus            RowStatus
}

midcomNatSessionAgentIndex OBJECT-TYPE
SYNTAX      MidcomAgentIndex
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION "Unique Identifier for an agent in the table"
::= { midcomNatSessionEntry 1 }

midcomNatSessionGroupId OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION "Group Identifier assignend to this
resource.

A value of 0 implies that the session does
not belong to a group membership.

::= { midcomNatSessionEntry 2 }

midcomNatSessionId OBJECT-TYPE
SYNTAX NatBindId
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Unique session Identifier assignend to this midcom bind resource. This identifier is same as the session identifier that is managed by the NAT middlebox.

::= { midcomNatSessionEntry 3 }

midcomNatSessionLifetime OBJECT-TYPE
SYNTAX TimeInterval
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Lifetime of the session. When this is set to 0 and GroupId is set to non-zero, the Lifetime of the GroupId is used to determine the lifetime of this resource.

::= { midcomNatSessionEntry 4 }

midcomNatSessionMaxIdletime OBJECT-TYPE
SYNTAX TimeInterval
MAX-ACCESS read-create
STATUS current
DESCRIPTION "MaxIdletime of the session. When this is set to 0 and GroupId is set to non-zero, the MaxIdletime of the GroupId is used to determine the MaxIdletime of this resource.

::= { midcomNatSessionEntry 5 }

midcomNatSessionIfIndex OBJECT-TYPE
SYNTAX InterfaceIndex
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Interface Index on which the bind is defined. This value may be set to 0 to mean any IP interface on the middlebox. This value may also be set to 0, when the middlebox has
just one interface on which midcom is configured.

::= { midcomNatSessionEntry 6 }

midcomNatSessionStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION "The status of this conceptual row. Objects in this row may be modified while the value of this object is active(1)."
REFERENCE "Textual Conventions for SMIPv2, Section 2
::= { midcomNatSessionEntry 7 }

--
-- midcomTransactions
-- The transaction Group
-- Transactions issued by the midcom agents
to the midcom MIB module.
--
--
-- Textual conventions used
--
--
MidcomInvocationStatus ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION "Allows invocation and status queries."
SYNTAX INTEGER {
neverInvoked(1),
performOperation(2),
inProgress(3),
success(4),
failure(5)
}

MidcomGroupCommand ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION "The choice of operations on groups.

add command:
Midcom agent uses the command to specify the group-identifiers and associated parameters it wishes to use during the Midcom session. In case of success, the GroupId is tracked by the midcom Module midcomGroupTable. No ill effect in case of failure.

delete command:
Midcom agent uses the command to remove a group-identifier from its list of valid group-ids. In case of success, the GroupId is deleted from the midcomGroupTable.

```
SYNTAX INTEGER {
   add(1),
   delete(2)
}
```

MidcomBindCommand ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION "The choice of operations on Nat Binds.

reserveBindInboundSrc, reserveBindInboundDst, reserveBindOutboundSrc, reserveBindOutboundDst
Reserve an address or port bind, given the interface and a src or dst endpoint in one of private address realm or public address realm.

reserveBindInboundSrcOrOutboundDst, reserveBindInboundDstOrOutboundSrc
Reserve an address or portBind, given the interface and a src or dst endpoint in one of private address realm or public address realm. Set the Bind to be bi-directional.

reserveBind2InboundSrc, reserveBind2InboundDst, reserveBind2OutboundSrc, reserveBind2OutboundDst
Reserve two port binds, given the interface index and a src or dst endpoint in one of private address realm or public address realm. The two ports assigned for the two port-binds are to be contiguous and assume oddity as specified in an oddity parameter. If the bind assigned turns out to be an address bind, one address
bind suffices independent of the port oddity requirement.

reserveBind2InboundSrcInboundDst,
reserveBind2OutboundSrcOutboundDst,

Reserve two binds as in a twice NAT, given the interface index and the session tuple in private realm or public realm.

SYNTAX INTEGER {
reserveBindInboundSrc,
reserveBindInboundDst,
reserveBindOutboundSrc,
reserveBindOutboundDst,
reserveBindInboundSrcOrOutboundDst,
reserveBindInboundDstOrOutboundSrc,
reserveBind2InboundSrc,
reserveBind2InboundDst,
reserveBind2OutboundSrc,
reserveBind2OutboundDst,
reserveBind2InboundSrcInboundDst,
reserveBind2OutboundSrcOutboundDst,
reserveBind2InboundSrcOrOutboundDst,
reserveBind2OutboundSrcOrOutboundDst,
}

MidcomNatSessionCommand ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
"The choice of commands on NAT sessions."

SYNTAX INTEGER {
createNatSession(1)
}

MidcomTransInOutFlags ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
"A BITS representation used to specify the relevant parameters for input during a command request (or) during a command response."

SYNTAX BITS {
privateAddrType (0),
privateSrcAddr (1),
privateSrcPort (2),
privateDstAddr (3),
privateDstPort (4),
globalAddrType (5),
globalSrcAddr (6),
globalSrcPort   (7),
globalDstAddr   (8),
globalDstPort   (9),
groupId         (10),
lifetime        (11),
maxIdletime     (12),
PrivateSrcBind  (13),
PrivateDstBind  (14)
}

MidcomSessionDirection ::= TEXTUAL-CONVENTION
  STATUS current
  DESCRIPTION
    "Describes the direction of a session specific to an
     interface."
  SYNTAX INTEGER {
    inbound(1),
    outbound(2)
  }

midcomTransGroupTable OBJECT-TYPE
  SYNTAX       SEQUENCE OF MidcomTransGroupEntry
  MAX-ACCESS   read-write
  STATUS       current
  DESCRIPTION  "This lists Group based transactions,
                one per each agent."
  ::= { midcomTransactions 1 }

midcomTransGroupEntry OBJECT-TYPE
  SYNTAX      MidcomTransGroupEntry
  MAX-ACCESS  not-accessible
  STATUS      current
  DESCRIPTION
    "Each entry pertains to a midcom agent carrying
     out a group based transaction. Midcom module will respond with Success or
     Failure, with an error code.

     In the case of success, the tuples specified in the
     transaction are entered into midcomGroupTable for
     later reference and parameter modification by the
     agent."
  INDEX   { midcomTransGroupAgentIndex }
  ::= { midcomTransGroupTable 1 }

MidcomTransGroupEntry ::= SEQUENCE {

Srisuresh
midcomTransGroupAgentIndex  MidcomAgentIndex,
midcomTransGroupMBResource  MidcomMBResource,
midcomTransGroupGroupId    Unsigned32,
midcomTransGroupLifetime   TimeInterval,
midcomTransGroupMaxIdletime TimeInterval,
midcomTransGroupCommand    MidcomGroupCommand,
midcomTransGroupStatus     MidcomInvocationStatus
}

midcomTransGroupAgentIndex OBJECT-TYPE
SYNTAX       MidcomAgentIndex
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "A unique Identifier for an Agent in the Table. This object is set when an agent reads the object midcomAgentIndexNext."
::= { midcomTransGroupEntry 1 }

midcomTransGroupMBResource OBJECT-TYPE
SYNTAX       MidcomMBResource
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  "Middlebox function specific resource type for which the GroupId is applicable."
::= { midcomTransGroupEntry 2 }

midcomTransGroupGroupId OBJECT-TYPE
SYNTAX       Unsigned32
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  "Group Identifier for which the Group operation is to be performed."
::= { midcomTransGroupEntry 3 }

midcomTransGroupLifetime OBJECT-TYPE
SYNTAX       TimeInterval
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  "Default Lifetime of the resources that are assigned this group Id. This field is required only during the add operation. This field is ignored during the delete operation."
::= { midcomTransGroupEntry 4 }

midcomTransGroupMaxIdletime OBJECT-TYPE
SYNTAX TimeInterval
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Default MaxIdletime of the resources that are assigned this group Id. This field is required to be filled only during the add operation. This field is ignored during the delete operation.
"

::= { midcomTransGroupEntry 5 }

midcomTransGroupCommand OBJECT-TYPE
SYNTAX MidcomGroupCommand
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This specifies the group command to be executed.
"

::= { midcomTransGroupEntry 6 }

midcomTransGroupStatus OBJECT-TYPE
SYNTAX MidcomInvocationStatus
MAX-ACCESS read-write
STATUS current
DESCRIPTION "Invocation status."

::= { midcomTransGroupEntry 7 }

midcomTransBindTable OBJECT-TYPE
SYNTAX SEQUENCE OF MidcomTransBindEntry
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This lists Bind based transactions, one per each agent."

::= { midcomTransactions 2 }

midcomTransBindEntry OBJECT-TYPE
SYNTAX MidcomTransBindEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "Each entry pertains to a midcom agent carrying out a BIND based transaction. Midcom module will respond with Success or Failure, with an error code.

In the case of success, there can be a maximum of two address or port binds returned. These binds are also entered into midcomBindTable
for later use by the midcom agents.

INDEX  { midcomTransBindAgentIndex }
  ::=  { midcomTransBindTable 1 }

MidcomTransBindEntry ::= SEQUENCE {
  midcomTransBindAgentIndex         MidcomAgentIndex,
  midcomTransBindCommand            MidcomBindCommand,
  midcomTransBindOddity             Unsigned32,
  midcomTransBindProtocol           NATProtocolType,
  midcomTransBindSessionDirection   MidcomSessionDirection,
  midcomTransBindIfIndex            InterfaceIndex,
  midcomTransBindInParms            MidcomTransInOutFlags,
  midcomTransBindOutParms           MidcomTransInOutFlags,
  midcomTransBindGroupId            Unsigned32,
  midcomTransBindLifetime           TimeInterval,
  midcomTransBindMaxIdleTime        TimeInterval,
  midcomTransBindPrivateAddrType    InetAddressType,
  midcomTransBindPrivateSrcAddr     InetAddress,
  midcomTransBindPrivateSrcPort     InetPortNumber,
  midcomTransBindPrivateDstAddr     InetAddress,
  midcomTransBindPrivateDstPort     InetPortNumber,
  midcomTransBindGlobalAddrType     InetAddressType,
  midcomTransBindGlobalSrcAddr      InetAddress,
  midcomTransBindGlobalSrcPort      InetPortNumber,
  midcomTransBindGlobalDstAddr      InetAddress,
  midcomTransBindGlobalDstPort      InetPortNumber,
  midcomTransBindPrivateSrcBindId   MidcomBindIdOrZero,
  midcomTransBindPrivateSrcBindMode MidcomBindMode,
  midcomTransBindPrivateDstBindId   MidcomBindIdOrZero,
  midcomTransBindPrivateDstBindMode MidcomBindMode,
  midcomTransBindStatus             MidcomInvocationStatus
}

midcomTransBindAgentIndex OBJECT-TYPE
SYNTAX       MidcomAgentIndex
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "A unique Identifier for an Agent in the Table.
              This object is set when an agent reads the object
              midcomAgentIndexNext."
  ::= { midcomTransBindEntry 1 }

midcomTransBindCommand  OBJECT-TYPE
SYNTAX MidcomBindCommand
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This specifies the bind command to be executed."

::= { midcomTransBindEntry 2 }

midcomTransBindOddity OBJECT-TYPE
SYNTAX INTEGER {
   oddityEnforce(1), -- Enforce oddity
   oddityNotRequired (2) -- Oddity not required.
}

::= { midcomTransBindEntry 3 }

midcomTransBindProtocol OBJECT-TYPE
SYNTAX NATProtocolType
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This specifies the protocol (TCP/UDP) of the session that requires the bind reservation."

::= { midcomTransBindEntry 4 }

midcomTransBindSessionDirection OBJECT-TYPE
SYNTAX MidcomSessionDirection
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This specifies the orientation of the session that requires the bind reservation."

::= { midcomTransBindEntry 5 }

midcomTransBindIfIndex OBJECT-TYPE
SYNTAX InterfaceIndex
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Interface Index for which the bind is being requested.

This value may be set to 0 to mean any
IP interface on the middlebox. This value may also be set to 0, when the middlebox has just one interface on which midcom is configured.

::= { midcomTransBindEntry 6 }

midcomTransBindInParms OBJECT-TYPE
SYNTAX MidcomTransInOutFlags
MAX-ACCESS read-write
STATUS current
DESCRIPTION "Lists the fields within the row that are filled by the requestor.

While the transaction allows for any or all of the end-points to be specified, typically, no more than one end-point should be defined. For Twice-Nat alone, two end-points must be specified.

::= { midcomTransBindEntry 7 }

midcomTransBindOutParms OBJECT-TYPE
SYNTAX MidcomTransInOutFlags
MAX-ACCESS read-write
STATUS current
DESCRIPTION "Lists the fields within the row that are filled by the middlebox in response to the bind request from agent.

While the transaction allows for any or all of the end-points to be filled, typically, no more than one end-point should be filled. For Twice-Nat alone, two end-points must be specified.

For oddity based port binds, the second bind is used to specify the second port bind.

::= { midcomTransBindEntry 8 }

midcomTransBindGroupId OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Group Identifier assigend to this bind resource."
A value of 0 implies that the bind is not assigned a group membership.

::= { midcomTransBindEntry 9 }

midcomTransBindLifetime OBJECT-TYPE
SYNTAX TimeInterval
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Individual Lifetime of the bind resource. When this is set to 0 and GroupId is set to non-zero, the Lifetime of the GroupId is used to determine the lifetime of this resource."

::= { midcomTransBindEntry 10 }

midcomTransBindMaxIdletime OBJECT-TYPE
SYNTAX TimeInterval
MAX-ACCESS read-create
STATUS current
DESCRIPTION "MaxIdletime of the Bind resource. When this is set to 0 and GroupId is set to non-zero, the MaxIdletime of the GroupId is used to determine the Maxidletime of this resource."

::= { midcomTransBindEntry 11 }

midcomTransBindPrivateAddrType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP address type in the private realm."

::= { midcomTransBindEntry 12 }

midcomTransBindPrivateSrcAddr OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP source address in the private realm. This is relevant if the agent refers a private realm address and the bind command is to find a bind for private realm source end point."

::= { midcomTransBindEntry 12 }
Internet-Draft                      Midcom MIB                      October 2003

::= { midcomTransBindEntry 13 }

midcomTransBindPrivateSrcPort   OBJECT-TYPE
SYNTAX       InetPortNumber
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  "IP source port in the private realm.
This is relevant if the agent refers a
private realm address and the bind command
is to find a bind for private realm
source end point.
"
 ::= { midcomTransBindEntry 14 }

midcomTransBindPrivateDstAddr   OBJECT-TYPE
SYNTAX       InetAddress
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  "IP destination address in the private realm.
This is relevant if the agent refers a
private realm address and the bind command
is to find a bind for private realm
destination end point.
"
 ::= { midcomTransBindEntry 15 }

midcomTransBindPrivateDstPort   OBJECT-TYPE
SYNTAX       InetPortNumber
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  "IP destination port in the private realm.
This is relevant if the agent refers a
private realm address and the bind command
is to find a bind for private realm
destination end point.
"
 ::= { midcomTransBindEntry 16 }

midcomTransBindGlobalAddrType   OBJECT-TYPE
SYNTAX       InetAddressType
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  "IP address type in the global address realm.
"
 ::= { midcomTransBindEntry 17 }

midcomTransBindGlobalSrcAddr   OBJECT-TYPE
SYNTAX       InetAddress

Srisuresh                                                      [Page 36]
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP source address in the global realm.
   This is relevant if the agent refers a
   global realm address and the bind command
   is to find a bind for global realm
   source end point.
"
::= { midcomTransBindEntry 18 }

midcomTransBindGlobalSrcPort OBJECT-TYPE
SYNTAX InetPortNumber
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP source port in the global realm.
   This is relevant if the agent refers a
   global realm address and the bind command
   is to find a bind for global realm
   source end point.
"
::= { midcomTransBindEntry 19 }

midcomTransBindGlobalDstAddr OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP destination address in the global realm.
   This is relevant if the agent refers a
   global realm address and the bind command
   is to find a bind for global realm
   destination end point.
"
::= { midcomTransBindEntry 20 }

midcomTransBindGlobalDstPort OBJECT-TYPE
SYNTAX InetPortNumber
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP destination port in the private realm.
   This is relevant if the agent refers a
   global realm address and the bind command
   is to find a bind for global realm
   destination end point.
"
::= { midcomTransBindEntry 21 }

midcomTransBindPrivateSrcBindId OBJECT-TYPE
SYNTAX MidcomBindIdOrZero
MAX-ACCESS  read-only
STATUS       current
DESCRIPTION  "This is the first Bind that will be generated in majority of the cases. This will be set to 0 in the case of symmetric NAT.
"
 ::= { midcomTransBindEntry 22 }

midcomTransBindPrivateSrcBindMode  OBJECT-TYPE
SYNTAX       MidcomBindMode,
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "This indicates whether PrivateSrcBind is address bind or port bind.
"
 ::= { midcomTransBindEntry 23 }

midcomTransBindPrivateDstBindId    OBJECT-TYPE
SYNTAX       MidcomBindIdOrZero
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "This is the second Bind that will be generated in the case of twice-NAT or oddity based 2 bind request. This will be set to 0 in the case of symmetric NAT.
"
 ::= { midcomTransBindEntry 24 }

midcomTransBindPrivateDstBindMode  OBJECT-TYPE
SYNTAX       MidcomBindMode
MAX-ACCESS   read-only
STATUS       current
DESCRIPTION  "This indicates whether PrivateDstBind is address bind or port bind.
"
 ::= { midcomTransBindEntry 25 }

midcomTransBindStatus   OBJECT-TYPE
SYNTAX       MidcomInvocationStatus
MAX-ACCESS   read-write
STATUS       current
DESCRIPTION  "Invocation status."
 ::= { midcomTransBindEntry 26 }

midcomTransNatSessionTable OBJECT-TYPE
SYNTAX        SEQUENCE OF MidcomTransNatSessionEntry
MAX-ACCESS    read-write
STATUS        current
DESCRIPTION   "This lists NatSession based transactions,
once per each agent."
 ::=    { midcomTransactions 3 }

midcomTransNatSessionEntry OBJECT-TYPE
SYNTAX        MidcomTransNatSessionEntry
MAX-ACCESS    not-accessible
STATUS        current
DESCRIPTION   "Each entry pertains to a midcom agent carrying
out a Nat session based transaction. Midcom module
will respond with Success or Failure, with an error code.

In the case of success, there can be a maximum
of two address or port binds returned. These
binds are entered into midcomBindTable
for later use by the midcom agents.

Further, the NatSession entry is included within
the midcomNatSession table.
"
INDEX    { midcomTransNatSessionAgentIndex }
 ::=    { midcomTransNatSessionTable 1 }

MidcomTransNatSessionEntry ::= SEQUENCE {
    midcomTransNatSessionAgentIndex        MidcomAgentIndex,
    midcomTransNatSessionCommand           MidcomNatSessionCommand,
    midcomTransNatSessionProtocol          NATProtocolType,
    midcomTransNatSessionSessionDirection  MidcomSessionDirection,
    midcomTransNatSessionIfIndex           InterfaceIndex,
    midcomTransNatSessionInParms           MidcomTransInOutFlags,
    midcomTransNatSessionOutParms          MidcomTransInOutFlags,
    midcomTransNatSessionGroupId           Unsigned32,
    midcomTransNatSessionLifetime          TimeInterval,
    midcomTransNatSessionMaxIdletime       TimeInterval,
    midcomTransNatSessionPrivateAddrType   InetAddressType,
    midcomTransNatSessionPrivateSrcAddr    InetAddress,
    midcomTransNatSessionPrivateSrcPort    InetPortNumber,
    midcomTransNatSessionPrivateDstAddr    InetAddress,
    midcomTransNatSessionPrivateDstPort    InetPortNumber,
    midcomTransNatSessionGlobalAddrType    InetAddressType,
    midcomTransNatSessionGlobalSrcAddr     InetAddress,
    midcomTransNatSessionGlobalSrcPort     InetPortNumber,
    midcomTransNatSessionGlobalDstAddr     InetAddress,
    midcomTransNatSessionGlobalDstPort     InetPortNumber,}
midcomTransNatSessionGlobalDstPort InetPortNumber,
midcomTransNatSessionPrivateSrcBindId MidcomBindIdOrZero,
midcomTransNatSessionPrivateDstBindId MidcomBindIdOrZero,
midcomTransNatSessionStatus MidcomInvocationStatus
}

midcomTransNatSessionAgentIndex OBJECT-TYPE
SYNTAX MidcomAgentIndex
MAX-ACCESS read-only
STATUS current
DESCRIPTION "A unique Identifier for an Agent in the Table. This object is set when an agent reads the object midcomAgentIndexNext."
::= { midcomTransNatSessionEntry 1 }

midcomTransNatSessionCommand OBJECT-TYPE
SYNTAX MidcomNatSessionCommand
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This specifies the NatSession command to be executed."
::= { midcomTransNatSessionEntry 2 }

midcomTransNatSessionProtocol OBJECT-TYPE
SYNTAX NATProtocolType
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This specifies the protocol (TCP/UDP) of the session."
::= { midcomTransNatSessionEntry 3 }

midcomTransNatSessionSessionDirection OBJECT-TYPE
SYNTAX MidcomSessionDirection
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This specifies the orientation of the session with reference to the interface index specified."
::= { midcomTransNatSessionEntry 4 }

midcomTransNatSessionIfIndex OBJECT-TYPE
SYNTAX InterfaceIndex
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Interface Index for which the NAT-Session is being requested.

This value may be set to 0 to mean any IP interface on the middlebox. This value may also be set to 0, when the middlebox has just one interface on which midcom is configured.

" :

::= { midcomTransNatSessionEntry 5 }

midcomTransNatSessionInParms OBJECT-TYPE
SYNTAX MidcomTransInOutFlags
MAX-ACCESS read-write
STATUS current
DESCRIPTION "Lists the fields within the row that are filled by the requestor.

While the transaction allows for any or all of the session parameters to be specified, typically, session parameters are filled in the private alone or in the public realm alone.

"

::= { midcomTransNatSessionEntry 6 }

midcomTransNatSessionOutParms OBJECT-TYPE
SYNTAX MidcomTransInOutFlags
MAX-ACCESS read-write
STATUS current
DESCRIPTION "Lists the fields within the row that are filled by the middlebox in response to the session request from agent.

While the transaction allows for any or all session parameters to be filled, typically, session parameters are filled in the private alone or in the public realm alone.

"

::= { midcomTransNatSessionEntry 7 }

midcomTransNatSessionGroupId OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Group Identifier assigend to this resource."
A value of 0 implies that the session is not assigned a group membership.

::= { midcomTransNatSessionEntry 8 }

midcomTransNatSessionLifetime OBJECT-TYPE
SYNTAX TimeInterval
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Individual Lifetime of the bind resource. When this is set to 0 and GroupId is set to non-zero, the Lifetime of the GroupId is used to determine the lifetime of this resource."

::= { midcomTransNatSessionEntry 9 }

midcomTransBindMaxIdletime OBJECT-TYPE
SYNTAX TimeInterval
MAX-ACCESS read-create
STATUS current
DESCRIPTION "MaxIdletime of the Bind resource. When this is set to 0 and GroupId is set to non-zero, the MaxIdletime of the GroupId is used to determine the Maxidletime of this resource."

::= { midcomTransNatSessionEntry 10 }

midcomTransBindPrivateAddrType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP address type in the private realm."

::= { midcomTransNatSessionEntry 11 }

midcomTransNatSessionPrivateSrcAddr OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP source address in the private realm. This is relevant if the agent refers a private realm session."

::= { midcomTransBindEntry 12 }
midcomTransNatSessionPrivateSrcPort OBJECT-TYPE
SYNTAX InetPortNumber
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP source port in the private realm. This is relevant if the agent refers a private realm based session."
::= { midcomTransNatSessionEntry 13 }

midcomTransNatSessionPrivateDstAddr OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP destination address in the private realm. This is relevant if the agent refers a private realm based session."
::= { midcomTransNatSessionEntry 14 }

midcomTransNatSessionPrivateDstPort OBJECT-TYPE
SYNTAX InetPortNumber
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP destination port in the private realm. This is relevant if the agent refers a private realm based session."
::= { midcomTransNatSessionEntry 15 }

midcomTransNatSessionGlobalAddrType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP address type in the global address realm."
::= { midcomTransNatSessionEntry 16 }

midcomTransBindGlobalSrcAddr OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS read-create
STATUS current
DESCRIPTION "IP source address in the global realm. This is relevant if the agent refers a global realm based session."
::= { midcomTransNatSessionEntry 17 }
midcomTransNatSessionGlobalSrcPort   OBJECT-TYPE
   SYNTAX       InetPortNumber
   MAX-ACCESS   read-create
   STATUS       current
   DESCRIPTION  "IP source port in the global realm. This is relevant if the agent refers a global realm based session.

   ::= { midcomTransNatSessionEntry 18 }

midcomTransNatSessionGlobalDstAddr   OBJECT-TYPE
   SYNTAX       InetAddress
   MAX-ACCESS   read-create
   STATUS       current
   DESCRIPTION  "IP destination address in the global realm. This is relevant if the agent refers a global realm based session.

   ::= { midcomTransNatSessionEntry 19 }

midcomTransNatSessionGlobalDstPort   OBJECT-TYPE
   SYNTAX       InetPortNumber
   MAX-ACCESS   read-create
   STATUS       current
   DESCRIPTION  "IP destination port in the private realm. This is relevant if the agent refers a global realm based session.

   ::= { midcomTransNatSessionEntry 20 }

midcomTransNatSessionPrivateSrcBindId   OBJECT-TYPE
   SYNTAX       MidcomBindIdOrZero
   MAX-ACCESS   read-create
   STATUS       current
   DESCRIPTION  "This is the first Bind that may be supplied by the agent. This BindId is the unique bindId for the midcom agent and is independent of what the NAt middlebox might have.

   This may be set to 0 in the case requestor does not have a BIND pre-assigned.

   ::= { midcomTransNatSessionEntry 21 }

midcomTransNatSessionPrivateDstBindId   OBJECT-TYPE
   SYNTAX       MidcomBindIdOrZero
   MAX-ACCESS   read-create
   STATUS       current
DESCRIPTION "This is the second Bind that may be supplied by the agent. This BindId is the unique bindId for the midcom agent and is independent of what the NAT middlebox might have. This may be set to 0 in the case requestor does not have a BIND pre-assigned."

::= { midcomTransNatSessionEntry 22 }

midcomTransNatSessionStatus OBJECT-TYPE
SYNTAX MidcomInvocationStatus
MAX-ACCESS read-write
STATUS current
DESCRIPTION "Invocation status."
::= { midcomTransNatSessionEntry 23 }

5. Security Considerations

The MIDCOM requirements [RFC3304] defines the general security requirements for the MIDCOM protocol. The SNMPv3 User-based Security Model (USM, [RFC2574]) satisfies those requirements. USM defines three standardized methods for providing authentication, confidentiality, and integrity. The method to use can be optionally chosen. The methods operate securely across untrusted domains. Additionally, USM has specific built-in mechanisms for preventing replay attacks including unique protocol engine IDs, timers and counters per engine and time windows for the validity of messages.

8. Acknowledgements

The author wishes to thank Wes Hardekar for kindly playing the role of MIB doctor on the raw initial versions of this document. The author also wishes to thank Dave Harrington for providing clarity on how and where to draw the line in defining the MIBs, given the interrelation between Midcom MIB and middlebox function MIBs. Lastly, the author wishes to thank Martin Stiemerling, Juergen Quittek, Tom Taylor and Mary Barnes for the numerous valuable e-mail discussions, phone conversations and feedback on the subject.

9. References

Normative References

RFC 3304, August, 2002.


Informative References


Authors’ Address

P. Srisuresh