

Network Working Group
Request for Comments: 2006
Category: Standards Track

D. Cong & M. Hamlen, Editors
Motorola
C. Perkins, Editor
IBM
October 1996

The Definitions of Managed Objects for IP Mobility Support using SMIV2

Status of this Memo

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

Abstract

This memo defines the Management Information Base (MIB) for use with network management protocols in TCP/IP-based internets. In particular, it describes managed objects used for managing the Mobile Node, Foreign Agent and Home Agent of the Mobile IP Protocol.

Table of Contents

1. The Network Management Framework	2
2. Objects	2
2.1 Object Definitions	2
3. Overview	2
3.1 Object Selection Criteria	2
3.2 Structure of the Mobile IP	3
3.3 MIB Groups	4
4. Definitions	5
5. Acknowledgements	49
6. Security Considerations	49
7. References	50
8. Chair's Address	51
9. Editors' Addresses	52

1. The SNMP Network Management Framework

The Internet-standard Network Management Framework presently consists of three major components. They are:

The SMI, described in RFC 1902 [1] - the mechanisms used for describing and naming objects for the purpose of management.

The MIB-II, STD 17, RFC 1213 [2] - the core set of managed objects for the Internet suite of protocols.

The protocol, RFC 1157 [3] and/or RFC 1905 [4], - the protocol for accessing managed objects.

The Framework permits new objects to be defined for the purpose of experimentation and evaluation.

2. Objects

2.1. Object Definitions

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) defined in the SMI. In particular, each object type is named by an OBJECT IDENTIFIER, an administratively assigned name. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the descriptor, to refer to the object type.

3. Overview

3.1. Object Selection Criteria

To be consistent with IAB directives and good engineering practice, the authors have applied some criteria to select managed objects for the Mobile IP Protocol.

(1) Partition management functionality among the Mobile Node, Home Agent, and Foreign Agent according to the partitioning seen in the Mobile IP Protocol.

(2) Require that objects be essential for either fault or configuration management.

(3) Limit the total number of objects.

(4) Exclude objects which are simply derivable from others in this or other MIBs.

3.2. Structure of the Mobile IP

This section describes the basic model of Mobile IP used in developing the Mobile IP MIB. This information should be useful to the implementor in understanding some of the basic design decisions of the MIB.

The Mobile IP Protocol introduces these new functional entities:

Mobile Node

A host or router that changes its point of attachment from one network or subnetwork to another. A mobile node may change its location without losing connectivity and without changing its IP address; it may continue to communicate with other Internet nodes at any location using its (constant) IP address, assuming link-layer connectivity to a point of attachment is available.

Home Agent

A router on a mobile node's home network which tunnels packets for delivery to the mobile node when it is away from home, and maintains current location information for the mobile node.

Foreign Agent

A router on a mobile node's visited network which provides routing services to the mobile node while registered. The foreign agent detunnels and delivers packets to the mobile node that were tunneled by the mobile node's home agent. For datagrams sent by a mobile node, the foreign agent may serve as a default router for registered mobile nodes.

This document specifies the objects used in managing these entities; namely, the Mobile Node, the Home Agent, and the Foreign Agent.

3.3. MIB Groups

Objects in this MIB are arranged into groups. Each group is organized as a set of related objects. The overall structure and the relationship between groups and the Mobile IP entities are shown below:

Groups	Mobile Node	Foreign Agent	Home Agent
mipSystemGroup	X	X	X
mipSecAssociationGroup	X	X	X
mipSecViolationGroup	X	X	X
mnSystemGroup	X		
mnDiscoveryGroup	X		
mnRegistrationGroup	X		
maAdvertisementGroup		X	X
faSystemGroup		X	
faAdvertisementGroup		X	
faRegistrationGroup		X	
haRegistrationGroup			X
haRegNodeCountersGroup			X

4. Definitions

```
MIP-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
Counter32, Gauge32, Integer32, IpAddress, experimental,  
MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE  
FROM SNMPv2-SMI
```

```
RowStatus, TruthValue, TimeStamp,  
TEXTUAL-CONVENTION
```

```
FROM SNMPv2-TC
```

```
MODULE-COMPLIANCE, OBJECT-GROUP
```

```
FROM SNMPv2-CONF;
```

```
mipMIB      MODULE-IDENTITY
```

```
    LAST-UPDATED      "9606040000Z"
```

```
    ORGANIZATION      "IETF Mobile IP Working Group"
```

```
    CONTACT-INFO
```

```
        "      David Cong
```

```
        Postal: Motorola
```

```
                1301 E. Algonquin Rd.
```

```
                Schaumburg, IL 60196
```

```
        Phone:  +1-847-576-1357
```

```
        Email:   cong@comm.mot.com"
```

```
    DESCRIPTION
```

```
        "The MIB Module for the Mobile IP."
```

```
    ::= { mib-2 44 }
```

```
mipMIBObjects  OBJECT IDENTIFIER ::= { mipMIB 1 }
```

```
-- Groups under mipMIBObjects
```

```
mipSystem      OBJECT IDENTIFIER ::= { mipMIBObjects 1 }
```

```
mipSecurity     OBJECT IDENTIFIER ::= { mipMIBObjects 2 }
```

```
mipMN           OBJECT IDENTIFIER ::= { mipMIBObjects 3 }
```

```
mipMA           OBJECT IDENTIFIER ::= { mipMIBObjects 4 }
```

```
mipFA           OBJECT IDENTIFIER ::= { mipMIBObjects 5 }
```

```
mipHA           OBJECT IDENTIFIER ::= { mipMIBObjects 6 }
```

```
mnSystem        OBJECT IDENTIFIER ::= { mipMN 1 }
```

```
mnDiscovery     OBJECT IDENTIFIER ::= { mipMN 2 }
```

```
mnRegistration  OBJECT IDENTIFIER ::= { mipMN 3 }
```

```
maAdvertisement  OBJECT IDENTIFIER ::= { mipMA 2 }
```

```
faSystem        OBJECT IDENTIFIER ::= { mipFA 1 }
```

```
faAdvertisement  OBJECT IDENTIFIER ::= { mipFA 2 }
```

```
faRegistration  OBJECT IDENTIFIER ::= { mipFA 3 }
```

```
haRegistration      OBJECT IDENTIFIER ::= { mipHA 3 }

-- Textual convention

RegistrationFlags   ::= TEXTUAL-CONVENTION
    STATUS          current
    DESCRIPTION
        "This data type is used to define the registration
        flags for Mobile IP registration extension:
        vjCompression
            -- Request to use VJ compression
        gre
            -- Request to use GRE
        minEnc
            -- Request to use minimal encapsulation
        decapsulationByMN
            -- Decapsulation by mobile node
        broadcastDatagram
            -- Request to receive broadcasts
        simultaneousBindings
            -- Request to retain prior binding(s)."
    SYNTAX          BITS {
        vjCompression(0),
        gre(1),
        minEnc(2),
        decapsulationbyMN(3),
        broadcastDatagram(4),
        simultaneousBindings(5)
    }

-- mipSystem Group

mipEntities OBJECT-TYPE
    SYNTAX          BITS {
        mobileNode(0),
        foreignAgent(1),
        homeAgent(2)
    }
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This object describes which Mobile IP entities are
        supported by this managed entity. The entity may
        support more than one Mobile IP entities. For example,
        the entity supports both Foreign Agent (FA) and Home
        Agent (HA). Therefore, bit 1 and bit 2 are set to 1
        for this object."
    ::= { mipSystem 1 }
```

```
mipEnable OBJECT-TYPE
    SYNTAX      INTEGER { enabled(1), disabled(2) }
    MAX-ACCESS   read-write
    STATUS       current
    DESCRIPTION
        "Indicates whether the Mobile IP protocol should be
        enabled for the managed entity. If it is disabled, the
        entity should disable both agent discovery and
        registration functions."
    ::= { mipSystem 2 }

mipEncapsulationSupported OBJECT-TYPE
    SYNTAX      BITS {
        ipInIp(0),
        gre(1),
        minEnc(2),
        other(3)
    }
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Encapsulation methods supported by the Mobile IP
        entity. The entity may support multiple encapsulation
        methods or none of them:
        ipInIp(0), -- IP Encapsulation within IP
        gre(1),   -- Generic Routing Encapsulation,
                   -- refers to RFC1701
        minEnc(2), -- Minimal Encapsulation within IP."
    ::= { mipSystem 3 }

-- mipSecurity Group

mipSecAssocTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MipSecAssocEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "A table containing Mobility Security Associations."
    ::= { mipSecurity 1 }

mipSecAssocEntry OBJECT-TYPE
    SYNTAX      MipSecAssocEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "One particular Mobility Security Association."
    INDEX      { mipSecPeerAddress, mipSecSPI }
    ::= { mipSecAssocTable 1 }
```

```
MipSecAssocEntry ::=
    SEQUENCE {
        mipSecPeerAddress IpAddress,
        mipSecSPI Unsigned32,
        mipSecAlgorithmType INTEGER,
        mipSecAlgorithmMode INTEGER,
        mipSecKey OCTET STRING,
        mipSecReplayMethod INTEGER
    }

mipSecPeerAddress OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The IP address of the peer entity with which this
        node shares the mobility security association."
    ::= { mipSecAssocEntry 1 }

mipSecSPI OBJECT-TYPE
    SYNTAX      Unsigned32 (0..4294967295)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The SPI is the 4-byte opaque index within the
        Mobility Security Association which selects the
        specific security parameters to be used to
        authenticate the peer, i.e. the rest of the variables
        in this MipSecAssocEntry."
    ::= { mipSecAssocEntry 2 }

mipSecAlgorithmType OBJECT-TYPE
    SYNTAX      INTEGER {
                    other(1),
                    md5(2)
                }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Type of security algorithm."
    ::= { mipSecAssocEntry 3 }

mipSecAlgorithmMode OBJECT-TYPE
    SYNTAX      INTEGER {
                    other(1),
                    prefixSuffix(2)
                }
    MAX-ACCESS  read-create
```



```
STATUS      current
DESCRIPTION
    "Security mode used by this algorithm."
 ::= { mipSecAssocEntry 4 }

mipSecKey OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE(16))
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The shared secret key for the security
         associations. Reading this object will always return
         zero length value."
    ::= { mipSecAssocEntry 5 }

mipSecReplayMethod OBJECT-TYPE
    SYNTAX      INTEGER {
                                other(1),
                                timestamps(2),
                                nonces(3)
                            }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The replay-protection method supported for this SPI
         within this Mobility Security Association."
    ::= { mipSecAssocEntry 6 }

-- Mobile IP security violation total counter

mipSecTotalViolations OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of security violations in the entity"
    ::= { mipSecurity 2 }

-- Mobile IP security violation table

mipSecViolationTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MipSecViolationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing information about security
         violations."
    ::= { mipSecurity 3 }
```

```
mipSecViolationEntry OBJECT-TYPE
    SYNTAX      MipSecViolationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Information about one particular security violation."
    INDEX       { mipSecViolatorAddress }
    ::= { mipSecViolationTable 1 }

MipSecViolationEntry ::=
    SEQUENCE {
        mipSecViolatorAddress IpAddress,
        mipSecViolationCounter Counter32,
        mipSecRecentViolationSPI Integer32,
        mipSecRecentViolationTime TimeStamp,
        mipSecRecentViolationIDLow Integer32,
        mipSecRecentViolationIDHigh Integer32,
        mipSecRecentViolationReason INTEGER
    }

mipSecViolatorAddress OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS  accessible-for-notify
    STATUS      current
    DESCRIPTION
        "Violator's IP address. The violator is not necessary
         in the mipSecAssocTable."
    ::= { mipSecViolationEntry 1 }

mipSecViolationCounter OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of security violations for this peer."
    ::= { mipSecViolationEntry 2 }

mipSecRecentViolationSPI OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "SPI of the most recent security violation for this
         peer. If the security violation is due to an
         identification mismatch, then this is the SPI from the
         Mobile-Home Authentication Extension. If the security
         violation is due to an invalid authenticator, then
         this is the SPI from the offending authentication"
```

```
        extension.  In all other cases, it should be set to
        zero."
 ::= { mipSecViolationEntry 3 }

mipSecRecentViolationTime OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Time of the most recent security violation for this
        peer."
 ::= { mipSecViolationEntry 4 }

mipSecRecentViolationIDLow  OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Low-order 32 bits of identification used in request or
        reply of the most recent security violation for this
        peer."
 ::= { mipSecViolationEntry 5 }

mipSecRecentViolationIDHigh OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "High-order 32 bits of identification used in request
        or reply of the most recent security violation for
        this peer."
 ::= { mipSecViolationEntry 6 }

mipSecRecentViolationReason OBJECT-TYPE
    SYNTAX      INTEGER {
                                noMobilitySecurityAssociation(1),
                                badAuthenticator(2),
                                badIdentifier(3),
                                badSPI(4),
                                missingSecurityExtension(5),
                                other(6)
                            }
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Reason for the most recent security violation for
        this peer."
 ::= { mipSecViolationEntry 7 }
```

```
-- mipMN Group
-- mipSystem Group

mnState OBJECT-TYPE
    SYNTAX      INTEGER {
                    home(1),
                    registered(2),
                    pending(3),
                    isolated(4),
                    unknown(5)
                }
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Indicates mobile node's state of Mobile IP:
         home,
            -- MN is connected to home network.
         registered,
            -- MN has registered on foreign network
         pending,
            -- MN has sent registration request and is
            waiting for the reply
         isolated,
            -- MN is isolated from network
         unknown
            -- MN can not determine its state."
    ::= { mnSystem 1 }

mnHomeAddress OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "An IP address that is assigned for an extended period
         of time to the mobile node. It remains unchanged
         regardless of the mobile node's current point of
         attachment."
    ::= { mnSystem 2 }

-- Mobile node's home agent list

mnHATable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MnHAEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
```

```
        "A table containing all of the mobile node's potential
        home agents."
 ::= { mnSystem 3 }

mnHAEEntry OBJECT-TYPE
    SYNTAX MnHAEEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Information for a particular Home Agent."
    INDEX { mnHAAAddress }
    ::= { mnHATable 1 }

MnHAEEntry ::= SEQUENCE {
    mnHAAAddress IpAddress,
    mnCurrentHA TruthValue,
    mnHASStatus RowStatus
}

mnHAAAddress OBJECT-TYPE
    SYNTAX IpAddress
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "IP address of mobile node's Home Agent."
    ::= { mnHAEEntry 1 }

mnCurrentHA OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Whether this home agent is the current home agent for
        the mobile node. If it is true, the mobile node is
        registered with that home agent."
    ::= { mnHAEEntry 2 }

mnHASStatus OBJECT-TYPE
    SYNTAX RowStatus
    MAX-ACCESS read-create
    STATUS current
    DESCRIPTION
        "The row status for this home agent entry. If the
        status is set to 'createAndGo' or 'active', then the
        mobile node can use mnHAAAddress as a valid candidate
        for a home agent. If the status is set to 'destroy',
        then the mobile node should delete this row, and
        deregister from that home agent."
```

```
::= { mnHAEEntry 3 }

mnFATable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MnFAEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing all foreign agents that the mobile
        node knows about and their corresponding COA (care-of
        address). This COA is an address of a foreign agent
        with which the mobile node is registered. The table is
        updated when advertisements are received by the mobile
        node. If an advertisement expires, its entry(s) should
        be deleted from the table. One foreign agent can
        provide more than one COA in its advertisements."
    ::= { mnDiscovery 1 }

mnFAEntry OBJECT-TYPE
    SYNTAX      MnFAEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "One pair of foreign agent IP address and COA for that
        foreign agent."
    INDEX { mnFAAddress, mnCOA }
    ::= { mnFATable 1 }

MnFAEntry ::= SEQUENCE {
    mnFAAddress IpAddress,
    mnCOA IpAddress
}

mnFAAddress OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Foreign agent's IP address."
    ::= { mnFAEntry 1 }

mnCOA OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "A care-of address being offered by this foreign agent
        or a co-located care-of address which the mobile node
        has associated with one of its own network"
```

```

        interfaces."
 ::= { mnFAEntry 2 }

-- Mobile node could store multiple agent advertisements, however,
-- only the most recently received agent advertisement information
-- is required to be made available to the manager station.

mnRecentAdvReceived OBJECT IDENTIFIER ::= { mnDiscovery 2 }

mnAdvSourceAddress OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The source IP address of the most recently received
        Agent Advertisement. This address could be the address
        of a home agent or a foreign agent."
    ::= { mnRecentAdvReceived 1 }

mnAdvSequence OBJECT-TYPE
    SYNTAX      INTEGER (0..65535)
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The sequence number of the most recently received
        advertisement. The sequence number ranges from 0 to
        0xffff. After the sequence number attains the value
        0xffff, it will roll over to 256."
    ::= { mnRecentAdvReceived 2 }

mnAdvFlags OBJECT-TYPE
    SYNTAX      BITS {
        vjCompression(0),
        gre(1),
        minEnc(2),
        foreignAgent(3),
        homeAgent(4),
        busy(5),
        regRequired(6)
    }
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The flags are contained in the 7th byte in the
        extension of the most recently received mobility agent
        advertisement:
        vjCompression
        -- Agent supports Van Jacobson compression

```

```
gre
    -- Agent offers Generic Routing Encapsulation
minEnc,
    -- Agent offers Minimal Encapsulation
foreignAgent,
    -- Agent is a Foreign Agent
homeAgent,
    -- Agent is a Home Agent
busy,
    -- Foreign Agent is busy
regRequired,
    -- FA registration is required."
 ::= { mnRecentAdvReceived 3 }

mnAdvMaxRegLifetime OBJECT-TYPE
    SYNTAX      INTEGER (0..65535)
    UNITS        "seconds"
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The longest lifetime in seconds that the agent is
        willing to accept in any registration request."
    ::= { mnRecentAdvReceived 4 }

mnAdvMaxAdvLifetime OBJECT-TYPE
    SYNTAX      INTEGER (0..65535)
    UNITS        "seconds"
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The maximum length of time that the Advertisement is
        considered valid in the absence of further
        Advertisements."
    REFERENCE
        "AdvertisementLifeTime in RFC1256."
    ::= { mnRecentAdvReceived 5 }

mnAdvTimeReceived OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The time at which the most recently received
        advertisement was received."
    ::= { mnRecentAdvReceived 6 }

-- Mobile Node Discovery Group Counter
```


mnSolicitationsSent OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Solicitation sent by the mobile node."

::= { mnDiscovery 3 }

mnAdvertisementsReceived OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of advertisements received by the mobile node."

::= { mnDiscovery 4 }

mnAdvsDroppedInvalidExtension OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of advertisements dropped by the mobile node due to both poorly formed extensions and unrecognized extensions with extension number in the range 0-127."

::= { mnDiscovery 5 }

mnAdvsIgnoredUnknownExtension OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of unrecognized extensions in the range 128-255 that were ignored by the mobile node."

::= { mnDiscovery 6 }

mnMoveFromHAToFA OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of times that the mobile node has decided to move from its home network to a foreign network."

::= { mnDiscovery 7 }

mnMoveFromFAToFA OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Number of times that the mobile node has decided to
    move from one foreign network to another foreign
    network."
 ::= { mnDiscovery 8 }

mnMoveFromFAToHA OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Number of times that the mobile node has decided to
        move from a foreign network to its home network."
    ::= { mnDiscovery 9 }

mnGratuitousARPsSend OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of Gratuitous ARPs sent by mobile node
        in order to clear out any stale ARP entries in the ARP
        caches of nodes on the home network."
    ::= { mnDiscovery 10 }

mnAgentRebootsDetected OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of agent reboots detected by the mobile
        node through sequence number of the advertisement."
    ::= { mnDiscovery 11 }

-- Mobile Node Registration Group

-- Registration table of mobile node

mnRegistrationTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MnRegistrationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing information about the mobile
        node's attempted registration(s).  The mobile node
```

```

        updates this table based upon Registration Requests
        sent and Registration Replies received in response to
        these requests.  Certain variables within this table
        are also updated if when Registration Requests are
        retransmitted."
 ::= { mnRegistration 1 }

mnRegistrationEntry OBJECT-TYPE
    SYNTAX      MnRegistrationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Information about one registration attempt."
    INDEX { mnRegAgentAddress, mnRegCOA }
 ::= { mnRegistrationTable 1 }

MnRegistrationEntry ::= SEQUENCE {
    mnRegAgentAddress IpAddress,
    mnRegCOA IpAddress,
    mnRegFlags RegistrationFlags,
    mnRegIDLow Integer32,
    mnRegIDHigh Integer32,
    mnRegTimeRequested Integer32,
    mnRegTimeRemaining Gauge32,
    mnRegTimeSent TimeStamp,
    mnRegIsAccepted TruthValue,
    mnCOAIsLocal TruthValue
}

mnRegAgentAddress OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "IP address of the agent as used in the destination
        IP address of the Registration Request.  The agent
        may be a home agent or a foreign agent."
 ::= { mnRegistrationEntry 1 }

mnRegCOA OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Care-of address for the registration."
 ::= { mnRegistrationEntry 2 }

mnRegFlags OBJECT-TYPE

```

```
SYNTAX      RegistrationFlags
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Registration flags sent by the mobile node. It is the
    second byte in the Mobile IP Registratation Request
    message."
 ::= { mnRegistrationEntry 3 }

mnRegIDLow OBJECT-TYPE
SYNTAX      Integer32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Low-order 32 bits of the Identification used in that
    registration by the mobile node."
 ::= { mnRegistrationEntry 4 }

mnRegIDHigh OBJECT-TYPE
SYNTAX      Integer32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "High-order 32 bits of the Identification used in that
    registration by the mobile node."
 ::= { mnRegistrationEntry 5 }

mnRegTimeRequested OBJECT-TYPE
SYNTAX      Integer32
UNITS       "seconds"
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "If the registration is pending, then this is the
    lifetime requested by the mobile node (in seconds).
    If the registration has been accepted, then this is
    the lifetime actually granted by the home agent in the
    reply."
 ::= { mnRegistrationEntry 6 }

mnRegTimeRemaining OBJECT-TYPE
SYNTAX      Gauge32
UNITS       "seconds"
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The number of seconds remaining until this
    registration expires. It has the same initial value
```

```
        as mnRegTimeRequested and is only valid if
        mnRegIsAccepted is TRUE."
 ::= { mnRegistrationEntry 7 }

mnRegTimeSent OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The time when the last (re-)transmission occurred."
 ::= { mnRegistrationEntry 8 }

mnRegIsAccepted OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "true(1) if the mobile node has received a
        Registration Reply indicating that service has been
        accepted; false(2) otherwise. false(2) implies that
        the registration is still pending."
 ::= { mnRegistrationEntry 9 }

mnCOAIsLocal OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "Whether the COA is local to (dynamically acquired by)
        the mobile node or not. If it is false(2), the COA is
        an address of the foreign agent."
 ::= { mnRegistrationEntry 10 }

-- Mobile Node Registration Group Counters

mnRegRequestsSent OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "Total number of registration requests sent by the
        mobile node. This does not include deregistrations
        (those with Lifetime equal to zero)."
 ::= { mnRegistration 2 }

mnDeRegRequestsSent OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS   read-only
```

```
STATUS      current
DESCRIPTION
    "Total number of deregistration requests sent by the
    mobile node (those with Lifetime equal to zero)."
```

```
::= { mnRegistration 3 }
```

```
mnRegRepliesRecieved OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of registration replies received by the
        mobile node in which the Lifetime is greater than
        zero."
```

```
::= { mnRegistration 4 }
```

```
mnDeRegRepliesRecieved OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of (de)registration replies received by
        the mobile node in which the Lifetime is equal to
        zero."
```

```
::= { mnRegistration 5 }
```

```
mnRepliesInvalidHomeAddress OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of replies with invalid home address for
        the mobile node."
```

```
::= { mnRegistration 6 }
```

```
mnRepliesUnknownHA OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of replies with unknown home agents
        (not in home agent table)."
```

```
::= { mnRegistration 7 }
```

```
mnRepliesUnknownFA OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
```

DESCRIPTION

"Total number of replies with unknown foreign agents if replies relayed through foreign agent."

::= { mnRegistration 8 }

mnRepliesInvalidID OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of replies with invalid Identification fields."

::= { mnRegistration 9 }

mnRepliesDroppedInvalidExtension OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Replies dropped by the mobile node due to both poorly formed extensions and unrecognized extensions with extension number in the range 0-127."

::= { mnRegistration 10 }

mnRepliesIgnoredUnknownExtension OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Replies that contained one or more unrecognized extensions in the range 128-255 that were ignored by the mobile node."

::= { mnRegistration 11 }

mnRepliesHAAAuthenticationFailure OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of replies without a valid Home Agent to Mobile Node authenticator."

::= { mnRegistration 12 }

mnRepliesFAAAuthenticationFailure OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of replies without a valid Foreign Agent to Mobile Node authenticator."

::= { mnRegistration 13 }

mnRegRequestsAccepted OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of registration requests accepted by the mobile node's home agent (Code 0 and Code 1)."

::= { mnRegistration 14 }

mnRegRequestsDeniedByHA OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of registration requests denied by mobile node's home agent (Sum of Code 128 through Code 191)."

::= { mnRegistration 15 }

mnRegRequestsDeniedByFA OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of registration requests denied by the foreign agent (Sum of Codes 64 through Code 127)."

::= { mnRegistration 16 }

mnRegRequestsDeniedByHADueToID OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Request denied by home agent due to identification mismatch."

::= { mnRegistration 17 }

mnRegRequestsWithDirectedBroadcast OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests sent by mobile


```
        node with a directed broadcast address in the home
        agent field."
 ::= { mnRegistration 18 }

-- MA Advertisement Group

-- Mobility agent advertisement configuration table

maAdvConfigTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MaAdvConfigEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing configurable advertisement
        parameters for all advertisement interfaces in
        the mobility agent."
    ::= { maAdvertisement 1 }

maAdvConfigEntry OBJECT-TYPE
    SYNTAX      MaAdvConfigEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Advertisement parameters for one advertisement
        interface."
    INDEX       { maInterfaceAddress }
    ::= { maAdvConfigTable 1 }

MaAdvConfigEntry ::= SEQUENCE {
    maInterfaceAddress IpAddress,
    maAdvMaxRegLifetime Integer32,
    maAdvPrefixLengthInclusion TruthValue,
    maAdvAddress IpAddress,
    maAdvMaxInterval Integer32,
    maAdvMinInterval Integer32,
    maAdvMaxAdvLifetime Integer32,
    maAdvResponseSolicitationOnly TruthValue,
    maAdvStatus RowStatus
}

maInterfaceAddress OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "IP address for advertisement interface."
    ::= { maAdvConfigEntry 1 }
```

maAdvMaxRegLifetime OBJECT-TYPE

SYNTAX Integer32 (0..65535)

UNITS "seconds"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The longest lifetime in seconds that mobility agent
is willing to accept in any Registration Request."

::= { maAdvConfigEntry 2 }

maAdvPrefixLengthInclusion OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"Whether the advertisement should include the Prefix-
Lengths Extension. If it is true, all advertisements
sent over this interface should include the
Prefix-Lengths Extension."

::= { maAdvConfigEntry 3 }

maAdvAddress OBJECT-TYPE

SYNTAX IpAddress

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The IP destination address to be used for
advertisements sent from the interface. The only
permissible values are the all-systems multicast
address (224.0.0.1) or the limited-broadcast address
(255.255.255.255)."

REFERENCE

"AdvertisementAddress in RFC1256."

::= { maAdvConfigEntry 4 }

maAdvMaxInterval OBJECT-TYPE

SYNTAX Integer32 (4..1800)

UNITS "seconds"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The maximum time in seconds between successive
transmissions of Agent Advertisements from this
interface."

REFERENCE

"MaxAdvertisementInterval in RFC1256."

::= { maAdvConfigEntry 5 }

```
maAdvMinInterval OBJECT-TYPE
    SYNTAX      Integer32 (3..1800)
    UNITS       "seconds"
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "The minimum time in seconds between successive
        transmissions of Agent Advertisements from this
        interface."
    REFERENCE
        "MinAdvertisementInterval in RFC1256."
    ::= { maAdvConfigEntry 6 }

maAdvMaxAdvLifetime OBJECT-TYPE
    SYNTAX      Integer32 (4..9000)
    UNITS       "seconds"
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "The time (in seconds) to be placed in the Lifetime
        field of the RFC 1256-portion of the Agent
        Advertisements sent over this interface."
    REFERENCE
        "AdvertisementLifetime in RFC1256."
    ::= { maAdvConfigEntry 7 }

maAdvResponseSolicitationOnly OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "The flag indicates whether the advertisement from
        that interface should be sent only in response to an
        Agent Solicitation message."
    DEFVAL      { false }
    ::= { maAdvConfigEntry 8 }

maAdvStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "The row status for the agent advertisement table. If
        this column status is 'active', the manager should not
        change any column in the row."
    ::= { maAdvConfigEntry 9 }

-- MA Advertisement Group Counters
```

maAdvertisementsSent OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of advertisements sent by the mobility agent."

::= { maAdvertisement 2 }

maAdvtsSentForSolicitation OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of advertisements sent by mobility agent in response to mobile node solicitations."

::= { maAdvertisement 3 }

maSolicitationsReceived OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of solicitations received by the mobility agent."

::= { maAdvertisement 4 }

-- Foreign Agent Group

-- Foreign Agent System Group

faCOATable OBJECT-TYPE

SYNTAX SEQUENCE OF FaCOAEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A table containing all of the care-of addresses (COAs) supported by the foreign agent. New entries can be added to the table. The order of entries in the faCOATable is also the order in which the COAs are listed in the Agent Advertisement."

::= { faSystem 1 }

faCOAEntry OBJECT-TYPE

SYNTAX FaCOAEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

```
        "Entry of COA"
INDEX { faSupportedCOA }
 ::= { faCOATable 1 }

FaCOAEntry ::=
    SEQUENCE {
        faSupportedCOA IpAddress,
        faCOAStatus    RowStatus
    }

faSupportedCOA OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "Care-of-address supported by this foreign agent."
    ::= { faCOAEntry 1 }

faCOAStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS   read-create
    STATUS       current
    DESCRIPTION
        "The row status for COA entry."
    ::= { faCOAEntry 2 }

-- Foreign Agent Advertisement Group
-- FA needs to implement MA Advertisement Group plus that group

faIsBusy OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Whether or not the foreign agent is too busy to
         accept additional registrations. If true(1), the agent
         is busy and any Agent advertisements sent from this
         agent should have the 'B' bit set to 1."
    ::= { faAdvertisement 1 }

faRegistrationRequired OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS   read-write
    STATUS       current
    DESCRIPTION
        "Whether or not this foreign agent requires
         registration even from those mobile nodes that have
         acquired their own, colocated care-of address.  If
```

```

        true(1), registration is required and any Agent
        Advertisements sent from this agent should have the
        'R' bit set to 1."
 ::= { faAdvertisement 2 }

-- Foreign Agent Registration Group

-- Foreign Agent Visitors List

faVisitorTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF FaVisitorEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing the foreign agent's visitor list.
        The foreign agent updates this table in response to
        registration events from mobile nodes."
    ::= { faRegistration 1 }

faVisitorEntry OBJECT-TYPE
    SYNTAX      FaVisitorEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Information for one visitor."
    INDEX       { faVisitorIPAddress }
    ::= { faVisitorTable 1 }

FaVisitorEntry ::= SEQUENCE {
    faVisitorIPAddress IpAddress,
    faVisitorHomeAddress IpAddress,
    faVisitorHomeAgentAddress IpAddress,
    faVisitorTimeGranted Integer32,
    faVisitorTimeRemaining Gauge32,
    faVisitorRegFlags RegistrationFlags,
    faVisitorRegIDLow Integer32,
    faVisitorRegIDHigh Integer32,
    faVisitorRegIsAccepted TruthValue
}

faVisitorIPAddress OBJECT-TYPE
    SYNTAX      IpAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Source IP address of visitor's Registration Request."
    ::= { faVisitorEntry 1 }

```

faVisitorHomeAddress OBJECT-TYPE

SYNTAX IpAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Home (IP) address of visiting mobile node."

::= { faVisitorEntry 2 }

faVisitorHomeAgentAddress OBJECT-TYPE

SYNTAX IpAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Home agent IP address for that visiting mobile node."

::= { faVisitorEntry 3 }

faVisitorTimeGranted OBJECT-TYPE

SYNTAX Integer32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The lifetime in seconds granted to the mobile node
for this registration. Only valid if
faVisitorRegIsAccepted is true(1)."

::= { faVisitorEntry 4 }

faVisitorTimeRemaining OBJECT-TYPE

SYNTAX Gauge32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of seconds remaining until the
registration is expired. It has the same initial value
as faVisitorTimeGranted, and is counted down by the
foreign agent."

::= { faVisitorEntry 5 }

faVisitorRegFlags OBJECT-TYPE

SYNTAX RegistrationFlags

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Registration flags sent by mobile node."

::= { faVisitorEntry 6 }

faVisitorRegIDLow OBJECT-TYPE

```
SYNTAX      Integer32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Low 32 bits of Identification used in that
    registration by the mobile node."
::= { faVisitorEntry 7 }
```

```
faVisitorRegIDHigh OBJECT-TYPE
SYNTAX      Integer32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "High 32 bits of Identification used in that
    registration by the mobile node."
::= { faVisitorEntry 8 }
```

```
faVisitorRegIsAccepted OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Whether the registration has been accepted or not. If
    it is false(2), this registration is still pending for
    reply."
::= { faVisitorEntry 9 }
```

-- Foreign Agent Registration Group Counters

```
faRegRequestsReceived OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of valid Registration Requests
    received."
::= { faRegistration 2 }
```

```
faRegRequestsRelayed OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Requests relayed to home
    agent by foreign agent."
::= { faRegistration 3 }
```

```
faReasonUnspecified OBJECT-TYPE
```



```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Requests denied by
    foreign agent -- reason unspecified (Code 64)."
```

::= { faRegistration 4 }

faAdmProhibited OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Requests denied by
    foreign agent -- administratively prohibited (Code
    65)."
```

::= { faRegistration 5 }

faInsufficientResource OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Requests denied by
    foreign agent -- insufficient resources (Code 66)."
```

::= { faRegistration 6 }

faMNAAuthenticationFailure OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Requests denied by
    foreign agent -- mobile node failed authentication
    (Code 67)."
```

::= { faRegistration 7 }

faRegLifetimeTooLong OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Requests denied by
    foreign agent -- requested lifetime too long (Code
    69)."
```

::= { faRegistration 8 }

faPoorlyFormedRequests OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Requests denied by
    foreign agent -- poorly formed request (Code 70)."
```

::= { faRegistration 9 }

faEncapsulationUnavailable OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Requests denied by
    foreign agent -- requested encapsulation unavailable
    (Code 72)."
```

::= { faRegistration 10 }

faVJCompressionUnavailable OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Requests denied by
    foreign agent -- requested Van Jacobson header
    compression unavailable (Code 73)."
```

::= { faRegistration 11 }

faHAUnreachable OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Requests denied by
    foreign agent -- home agent unreachable (Codes
    80-95)."
```

::= { faRegistration 12 }

faRegRepliesRecieved OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of well-formed Registration Replies
    received by foreign agent."
```

::= { faRegistration 13 }

faRegRepliesRelayed OBJECT-TYPE

```
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of valid Registration Replies relayed to
    the mobile node by foreign agent."
 ::= { faRegistration 14 }

faHAAAuthenticationFailure OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Replies denied by
    foreign agent -- home agent failed authentication
    (Code 68)."
```

```
 ::= { faRegistration 15 }

faPoorlyFormedReplies OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Total number of Registration Replies denied by
    foreign agent -- poorly formed reply (Code 71)."
```

```
 ::= { faRegistration 16 }

-- Home Agent Group

-- Home Agent Registration Group

-- Home agent mobility binding list

haMobilityBindingTable OBJECT-TYPE
SYNTAX      SEQUENCE OF HaMobilityBindingEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "A table containing the home agent's mobility binding
    list.  The home agent updates this table in response
    to registration events from mobile nodes."
```

```
 ::= { haRegistration 1 }

haMobilityBindingEntry OBJECT-TYPE
SYNTAX      HaMobilityBindingEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
```

```

        "An entry on the mobility binding list."
INDEX   { haMobilityBindingMN, haMobilityBindingCOA }
 ::= { haMobilityBindingTable 1 }

HaMobilityBindingEntry ::= SEQUENCE {
    haMobilityBindingMN      IPAddress,
    haMobilityBindingCOA     IPAddress,
    haMobilityBindingSourceAddress  IPAddress,
    haMobilityBindingRegFlags  RegistrationFlags,
    haMobilityBindingRegIDLow Integer32,
    haMobilityBindingRegIDHigh Integer32,
    haMobilityBindingTimeGranted Integer32,
    haMobilityBindingTimeRemaining Gauge32
}

haMobilityBindingMN      OBJECT-TYPE
    SYNTAX      IPAddress
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "Mobile node's home (IP) address."
    ::= { haMobilityBindingEntry 1 }

haMobilityBindingCOA     OBJECT-TYPE
    SYNTAX      IPAddress
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "Mobile node's care-of-address. One mobile node can
        have multiple bindings with different
        care-of-addresses."
    ::= { haMobilityBindingEntry 2 }

haMobilityBindingSourceAddress  OBJECT-TYPE
    SYNTAX      IPAddress
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "IP source address of the Registration Request as
        received by the home agent. Will be either a mobile
        node's co-located care-of address or an address of the
        foreign agent."
    ::= { haMobilityBindingEntry 3 }

haMobilityBindingRegFlags OBJECT-TYPE
    SYNTAX      RegistrationFlags
    MAX-ACCESS   read-only
    STATUS      current

```

DESCRIPTION

"Registration flags sent by mobile node."

::= { haMobilityBindingEntry 4 }

haMobilityBindingRegIDLow OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Low 32 bits of Identification used in that binding by the mobile node."

::= { haMobilityBindingEntry 5 }

haMobilityBindingRegIDHigh OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"High 32 bits of Identification used in that binding by the mobile node."

::= { haMobilityBindingEntry 6 }

haMobilityBindingTimeGranted OBJECT-TYPE

SYNTAX Integer32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The lifetime in seconds granted to the mobile node for this registration."

::= { haMobilityBindingEntry 7 }

haMobilityBindingTimeRemaining OBJECT-TYPE

SYNTAX Gauge32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of seconds remaining until the registration is expired. It has the same initial value as haMobilityBindingTimeGranted, and is counted down by the home agent."

::= { haMobilityBindingEntry 8 }

-- Home Agent Registration Group Counters

-- Home agent registration Counters per node

```
haCounterTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF HaCounterEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing registration statistics for all
        mobile nodes authorized to use this home agent."
    ::= { haRegistration 2 }

haCounterEntry OBJECT-TYPE
    SYNTAX      HaCounterEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Registration statistics for one mobile node."
    INDEX       { haMobilityBindingMN }
    ::= { haCounterTable 1 }

HaCounterEntry ::= SEQUENCE {
    haServiceRequestsAccepted Counter32,
    haServiceRequestsDenied Counter32,
    haOverallServiceTime Gauge32,
    haRecentServiceAcceptedTime TimeStamp,
    haRecentServiceDeniedTime TimeStamp,
    haRecentServiceDeniedCode INTEGER
}

haServiceRequestsAccepted OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of service requests for the mobile node
        accepted by the home agent (Code 0 + Code 1)."
    ::= { haCounterEntry 2 }

haServiceRequestsDenied OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of service requests for the mobile node
        denied by the home agent (sum of all registrations
        denied with Code 128 through Code 159)."
    ::= { haCounterEntry 3 }

haOverallServiceTime OBJECT-TYPE
    SYNTAX      Gauge32
```

```

UNITS          "seconds"
MAX-ACCESS     read-only
STATUS         current
DESCRIPTION    "Overall service time (in seconds) that has
               accumulated for the mobile node since the home agent
               last rebooted."
 ::= { haCounterEntry 4 }

haRecentServiceAcceptedTime  OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION  "The time at which the most recent Registration
               Request was accepted by the home agent for this mobile
               node."
 ::= { haCounterEntry 5 }

haRecentServiceDeniedTime    OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION  "The time at which the most recent Registration
               Request was denied by the home agent for this mobile
               node."
 ::= { haCounterEntry 6 }

haRecentServiceDeniedCode    OBJECT-TYPE
    SYNTAX      INTEGER {
        reasonUnspecified(128),
        admProhibited(129),
        insufficientResource(130),
        mnAuthenticationFailure(131),
        faAuthenticationFailure(132),
        idMismatch(133),
        poorlyFormedRequest(134),
        tooManyBindings(135),
        unknownHA(136)
    }
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION  "The Code indicating the reason why the most recent
               Registration Request for this mobile node was rejected
               by the home agent."
 ::= { haCounterEntry 7 }
```

-- Home agent registration Counters for all mobile nodes.

haRegistrationAccepted OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests accepted by
home agent (Code 0)."

::= { haRegistration 3 }

haMultiBindingUnsupported OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests accepted by
home agent -- simultaneous mobility bindings
unsupported (Code 1)."

::= { haRegistration 4 }

haReasonUnspecified OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests denied by home
agent -- reason unspecified (Code 128)."

::= { haRegistration 5 }

haAdmProhibited OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests denied by home
agent -- administratively prohibited (Code 129)."

::= { haRegistration 6 }

haInsufficientResource OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests denied by home
agent -- insufficient resources (Code 130)."

::= { haRegistration 7 }

haMNAuthenticationFailure OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests denied by home agent -- mobile node failed authentication (Code 131)."

::= { haRegistration 8 }

haFAAuthenticationFailure OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests denied by home agent -- foreign agent failed authentication (Code 132)."

::= { haRegistration 9 }

haIDMismatch OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests denied by home agent -- Identification mismatch (Code 133)."

::= { haRegistration 10 }

haPoorlyFormedRequest OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests denied by home agent -- poorly formed request (Code 134)."

::= { haRegistration 11 }

haTooManyBindings OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Total number of Registration Requests denied by home agent -- too many simultaneous mobility bindings (Code 135)."

::= { haRegistration 12 }

```
haUnknownHA      OBJECT-TYPE
    SYNTAX          Counter32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Total number of Registration Requests denied by home
        agent -- unknown home agent address (Code 136)."
```

::= { haRegistration 13 }

```
haGratuitiousARPsSent OBJECT-TYPE
    SYNTAX          Counter32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Total number of gratuition ARPs sent by the home
        agent on behalf of mobile nodes."
```

::= { haRegistration 14 }

```
haProxyARPsSent   OBJECT-TYPE
    SYNTAX          Counter32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Total number of proxy ARPs sent by the home agent on
        behalf of mobile nodes."
```

::= { haRegistration 15 }

```
haRegRequestsReceived OBJECT-TYPE
    SYNTAX          Counter32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Total number of Registration Requests received by
        home agent."
```

::= { haRegistration 16 }

```
haDeRegRequestsReceived OBJECT-TYPE
    SYNTAX          Counter32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Total number of Registration Requests received by the
        home agent with a Lifetime of zero (requests to
        deregister)."
```

::= { haRegistration 17 }

```
haRegRepliesSent  OBJECT-TYPE
    SYNTAX          Counter32
```

```

MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
    "Total number of Registration Replies sent by the home
    agent."
 ::= { haRegistration 18 }

haDeRegRepliesSent OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Total number of Registration Replies sent by the home
        agent in response to requests to deregister."
    ::= { haRegistration 19 }

mipMIBNotificationPrefix    OBJECT IDENTIFIER ::= { mipMIB 2 }

mipMIBNotifications    OBJECT IDENTIFIER ::=
    { mipMIBNotificationPrefix 0 }

mipAuthFailure NOTIFICATION-TYPE
    OBJECTS      {
        mipSecViolatorAddress,
        mipSecRecentViolationSPI,
        mipSecRecentViolationIDLow,
        mipSecRecentViolationIDHigh,
        mipSecRecentViolationReason
    }

    STATUS      current
    DESCRIPTION
        "The mipAuthFailure indicates that the Mobile IP
        entity has an authentication failure when it validates
        the mobile Registration Request or Reply.
        Implementation of this trap is optional."
    ::= { mipMIBNotifications 1 }

mipMIBConformance OBJECT IDENTIFIER ::= { mipMIB 3 }

mipGroups          OBJECT IDENTIFIER ::= { mipMIBConformance 1 }
mipCompliances      OBJECT IDENTIFIER ::= { mipMIBConformance 2 }

-- compliance statements

mipCompliance      MODULE-COMPLIANCE
    STATUS          current
    DESCRIPTION

```

```
"The compliance statement for SNMPv2 entities which
implement the Mobile IP MIB."
MODULE
  MANDATORY-GROUPS { mipSystemGroup }

  GROUP      mipSecAssociationGroup
  DESCRIPTION
    "This group is mandatory for Mobile IP entities (MN,
    FA, and HA) which support security associations.
    Mobile Nodes and Home Agents must implement this
    group. Foreign Agents must implement this group if
    they maintain any security associations."

  GROUP      mipSecViolationGroup
  DESCRIPTION
    "This group is mandatory for Mobile IP entities (MN,
    FA, and HA) that can log security violations."

  GROUP      mnSystemGroup
  DESCRIPTION
    "This group is mandatory for mobile node."

  GROUP      mnDiscoveryGroup
  DESCRIPTION
    "This group is mandatory for mobile nodes which
    implement the Agent Discovery function."

  GROUP      mnRegistrationGroup
  DESCRIPTION
    "This group is mandatory for mobile nodes."

  GROUP      maAdvertisementGroup
  DESCRIPTION
    "This group is mandatory for the mobility agents (HA
    and FA) since they must implement Agent
    Advertisement."

  GROUP      faSystemGroup
  DESCRIPTION
    "This group is mandatory for foreign agents."

  GROUP      faAdvertisementGroup
  DESCRIPTION
    "This group is mandatory for foreign agents."

  GROUP      faRegistrationGroup
  DESCRIPTION
    "This group is mandatory for foreign agents."
```

```
GROUP      haRegistrationGroup
DESCRIPTION
    "This group is mandatory for home agents."

GROUP      haRegNodeCountersGroup
DESCRIPTION
    "This group is mandatory for home agents which log
    registration counters for each individual mobile
    node."

GROUP      mipSecNotificationsGroup
DESCRIPTION
    "This group is mandatory for Mobile IP entities (MN,
    FA, and HA) that can report the security violations."

 ::= { mipCompliances 1 }

-- Units of conformance

mipSystemGroup      OBJECT-GROUP
    OBJECTS      { mipEntities, mipEnable, mipEncapsulationSupported }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing the basic Mobile IP
        entity's management information."
    ::= { mipGroups 1 }

mipSecAssociationGroup OBJECT-GROUP
    OBJECTS      { mipSecAlgorithmType, mipSecAlgorithmMode,
        mipSecKey, mipSecReplayMethod }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing the management
        information for security associations of Mobile IP
        entities."
    ::= { mipGroups 2 }

mipSecViolationGroup      OBJECT-GROUP
    OBJECTS      { mipSecTotalViolations,
        mipSecViolationCounter, mipSecRecentViolationSPI,
        mipSecRecentViolationTime,
        mipSecRecentViolationIDLow,
        mipSecRecentViolationIDHigh,
        mipSecRecentViolationReason }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing the management
```

```

        information for security violation logging of Mobile
        IP entities."
 ::= { mipGroups 3 }

mnSystemGroup      OBJECT-GROUP
  OBJECTS          { mnState, mnCurrentHA, mnHomeAddress,
                     mnHAStatus }
  STATUS            current
  DESCRIPTION
    "A collection of objects providing the basic
    management information for mobile nodes."
 ::= { mipGroups 4 }

mnDiscoveryGroup    OBJECT-GROUP
  OBJECTS          { mnFAAddress, mnCOA, mnAdvSourceAddress,
                     mnAdvSequence, mnAdvFlags, mnAdvMaxRegLifetime,
                     mnAdvMaxAdvLifetime, mnAdvTimeReceived,
                     mnSolicitationsSent, mnAdvertisementsReceived,
                     mnAdvsDroppedInvalidExtension,
                     mnAdvsIgnoredUnknownExtension, mnMoveFromHAToFA,
                     mnMoveFromFAToFA, mnMoveFromFAToHA,
                     mnGratuitousARPsSend, mnAgentRebootsDecteded }
  STATUS            current
  DESCRIPTION
    "A collection of objects providing management
    information for the Agent Discovery function within a
    mobile node."
 ::= { mipGroups 5 }

mnRegistrationGroup OBJECT-GROUP
  OBJECTS          { mnRegAgentAddress, mnRegCOA, mnRegFlags, mnRegIDLow,
                     mnRegIDHigh, mnRegTimeRequested, mnRegTimeRemaining,
                     mnRegTimeSent, mnRegIsAccepted, mnCOAIsLocal,
                     mnRegRequestsSent, mnRegRepliesRecieved,
                     mnDeRegRequestsSent, mnDeRegRepliesRecieved,
                     mnRepliesInvalidHomeAddress, mnRepliesUnknownHA,
                     mnRepliesUnknownFA, mnRepliesInvalidID,
                     mnRepliesDroppedInvalidExtension,
                     mnRepliesIgnoredUnknownExtension,
                     mnRepliesHAAAuthenticationFailure,
                     mnRepliesFAAAuthenticationFailure,
                     mnRegRequestsAccepted, mnRegRequestsDeniedByHA,
                     mnRegRequestsDeniedByFA,
                     mnRegRequestsDeniedByHADueToID,
                     mnRegRequestsWithDirectedBroadcast }
  STATUS            current
  DESCRIPTION
    "A collection of objects providing management

```

```
        information for the registration function within a
        mobile node."
 ::= { mipGroups 6 }

maAdvertisementGroup      OBJECT-GROUP
  OBJECTS      { maAdvMaxRegLifetime,
                  maAdvPrefixLengthInclusion, maAdvAddress,
                  maAdvMaxInterval, maAdvMinInterval,
                  maAdvMaxAdvLifetime,
                  maAdvResponseSolicitationOnly, maAdvStatus,
                  maAdvertisementsSent, maAdvsSentForSolicitation,
                  maSolicitationsReceived }
  STATUS      current
  DESCRIPTION
    "A collection of objects providing management
    information for the Agent Advertisement function
    within mobility agents."
 ::= { mipGroups 7 }

faSystemGroup            OBJECT-GROUP
  OBJECTS      { faCOAStatus}
  STATUS      current
  DESCRIPTION
    "A collection of objects providing the basic
    management information for foreign agents."
 ::= { mipGroups 8 }

faAdvertisementGroup      OBJECT-GROUP
  OBJECTS      { faIsBusy, faRegistrationRequired }
  STATUS      current
  DESCRIPTION
    "A collection of objects providing supplemental
    management information for the Agent Advertisement
    function within a foreign agent."
 ::= { mipGroups 9 }

faRegistrationGroup      OBJECT-GROUP
  OBJECTS      { faVisitorIPAddress, faVisitorHomeAddress,
                  faVisitorHomeAgentAddress, faVisitorTimeGranted,
                  faVisitorTimeRemaining, faVisitorRegFlags,
                  faVisitorRegIDLow, faVisitorRegIDHigh,
                  faVisitorRegIsAccepted, faRegRequestsReceived,
                  faRegRequestsRelayed, faReasonUnspecified,
                  faAdmProhibited, faInsufficientResource,
                  faMNAuthenticationFailure, faRegLifetimeTooLong,
                  faPoorlyFormedRequests,
                  faEncapsulationUnavailable,
                  faVJCompressionUnavailable, faHAUnreachable,
```

```

        faRegRepliesRecieved, faRegRepliesRelayed,
        faHAAAuthenticationFailure, faPoorlyFormedReplies }
STATUS      current
DESCRIPTION
    "A collection of objects providing management
    information for the registration function within a
    foreign agent."
 ::= { mipGroups 10 }

haRegistrationGroup    OBJECT-GROUP
OBJECTS    { haMobilityBindingMN, haMobilityBindingCOA,
             haMobilityBindingSourceAddress,
             haMobilityBindingRegFlags,
             haMobilityBindingRegIDLow,
             haMobilityBindingRegIDHigh,
             haMobilityBindingTimeGranted,
             haMobilityBindingTimeRemaining,
             haRegistrationAccepted, haMultiBindingUnsupported,
             haReasonUnspecified, haAdmProhibited,
             haInsufficientResource, haMNAAuthenticationFailure,
             haFAAuthenticationFailure, haIDMismatch,
             haPoorlyFormedRequest, haTooManyBindings,
             haUnknownHA, haGratuitiousARPsSent,
             haProxyARPsSent, haRegRequestsReceived,
             haDeRegRequestsReceived, haRegRepliesSent,
             haDeRegRepliesSent }
STATUS      current
DESCRIPTION
    "A collection of objects providing management
    information for the registration function within a
    home agent."
 ::= { mipGroups 11 }

haRegNodeCountersGroup OBJECT-GROUP
OBJECTS    { haServiceRequestsAccepted,
             haServiceRequestsDenied, haOverallServiceTime,
             haRecentServiceAcceptedTime,
             haRecentServiceDeniedTime,
             haRecentServiceDeniedCode }
STATUS      current
DESCRIPTION
    "A collection of objects providing management
    information for counters related to the registration
    function within a home agent."
 ::= { mipGroups 12 }

mipSecNotifcationsGroup NOTIFICATION-GROUP
NOTIFICATIONS { mipAuthFailure }

```



```
STATUS      current
DESCRIPTION
    "The notification related to security violations."
 ::= { mipGroups 13 }
```

END

5. Acknowledgments

This document was produced by the Mobile IP working group. The editors wish to thank Bob Stewart (Cisco Systems), for his help in converting from SNMPv1 to SNMPv2. We also want to thank Jim Solomon, for his encouragement, patience, and help. Thanks to Fredrick Tarberg and Fredrik Broman (KTH) for their initial efforts in defining a Mobile IP MIB. Thanks to Frank Kastenholz (FTP Software) for his comments on the initial MIB from KTH. Thanks to Gerald Maguire (KTH) for his comments on the first version of this MIB. Thanks to Mike Roels (Motorola) for his help in testing this MIB.

6. Security Considerations

The Mobile IP MIB affords the network operator the ability to configure and control the Mobile IP links of a particular system, including the Mobile IP authentication protocols, and shared secret key. This represents a security risk.

These risks are addressed in the following manners:

- (1) All variables which represent a significant security risk are placed in separate MIB Groups. By providing Agent Capability Statements, the implementor of the MIB may elect not to implement these groups.
- (2) The MIB allows the manager station to create the security association for Mobile IP entities. However, the agent should always return 0 length octet string when the manager station retrieves the shared security key in the mipSecAssocTable. In this way, the Mobile IP entities can prevent the key leaking from SNMP GET, GET-NEXT, or GET-BULK requests.
- (3) The MIB defines a trap for Mobile IP entities to send a notification to the manager station if there is a security violation. In this way, the operator can notice the source of an intruder.
- (4) The MIB also defines a table to log the security violations in the Mobile IP entities. The manager station can retrieve this log to analyze the security violation instances in the

system.

Thus, in order to preserve the integrity, security and privacy of the Mobile IP security features, an implementation SHOULD allow access to this MIB only via SNMPv2 and with other security enhancement such as SNMPv2Sec. The other way to access this information is in concert with the IP security protocols (IP Authentication Header and IP Encapsulating Security Payload).

7.0 References

- [1] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Structure of Management Information for version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1902, January 1996.
- [2] McCloghrie, K., and M. Rose, Editors, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II", STD 17, RFC 1213, March 1991.
- [3] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol", RFC 1157, May 1990.
- [4] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Protocol Operations for version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1905, January 1996.
- [5] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Management Information Base for version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1907, January 1996.
- [6] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Textual Conventions for version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1903, January 1996.
- [7] Solomon J., "Mobile IP Protocol Applicability Statement", RFC 2005, October 1996.
- [8] Perkins C., "IP Mobility Support", RFC 2002, October 1996.
- [9] Perkins C., "IP Encapsulation within IP", RFC 2003, October 1996.
- [10] Perkins C., "Minimal Encapsulation within IP", RFC 2004, October 1996.

- [11] Hanks S. et. al., "Generic Routing Encapsulation (GRE)", RFC 1701, October 1994.
- [12] Deering, S., "ICMP Router Discovery Messages", RFC 1256, September 1991.
- [13] Atkinson, R., "IP Authentication Header", RFC 1826, August 1995.
- [14] Atkinson, R., "IP Encapsulating Security Payload (ESP)", RFC 1827, August 1995.

8. Chair's Address

The working group can be contacted via the current chair:

Jim Solomon
Motorola, Inc.
1301 E. Algonquin Rd.
Schaumburg, IL 60196

Work: +1-847-576-2753
Fax: +1-847-576-3240
EMail: solomon@comm.mot.com

9. Editors' Addresses

Questions about this memo can also be directed to:

David Cong
Room 3149
Motorola
1301 East Algonquin Rd.
Schaumburg, IL 60196

Work: +1-847-576-1357
Fax: +1-847-538-3472
EMail: cong@comm.mot.com

Mark Hamlen
Room 4413
Motorola
1301 East Algonquin Rd.
Schaumburg, IL 60196

Work: +1-847-576-0346
Fax: +1-847-538-6150
EMail: hamlen@comm.mot.com

Charles Perkins
Room J1-A25
T. J. Watson Research Center
IBM Corporation
30 Saw Mill River Rd.
Hawthorne, NY 10532

Work: +1-914-784-7350
Fax: +1-914-784-7007
EMail: perk@watson.ibm.com

