**IEEE P802.21m  
Media Independent Services Framework Project**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DCN 21-14-0067-00-REVP: 802.21-2008, 802.21a, and 802.21b text to be included** | | | | |
| **Date:** March 20, 2014 | | | | |
| **Author(s):** | | | | |
| **Name** | **Affiliation** | **Address** | **Phone** | **Email** |
| Charlie Perkins | Futurewei | Santa Clara, CA 95050 |  | charliep@computer.org |

Abstract

This document contains text extracted from 802.21-2008, 802.21a, and 802.21b which is proposed to be included in the document developed within 802.21m (REVP project).

P802.21™/D  
  
Draft Standard for Local and metropolitan area networks  
  
Part 21: Media Independent Services Framework

Sponsor

LAN/MAN Standards Committee

of the

IEEE Computer Society

Approved

IEEE-SA Standards Board

Copyright © 2013 by the Institute of Electrical and Electronics Engineers, Inc.

Three Park Avenue

New York, New York 10016-5997, USA

All rights reserved.

This document is an unapproved draft of a proposed IEEE Standard. As such, this document is subject to change. USE AT YOUR OWN RISK! Because this is an unapproved draft, this document must not be utilized for any conformance/compliance purposes. Permission is hereby granted for IEEE Standards Comittee participants to reproduce this document for purposes of standardization consideration. Prior to adoption of this document, in whole or in part, by another standards development organization, permission must first be obtained from the IEEE Standards Activities Department ([stds.ipr@ieee.org](mailto:stds.ipr@ieee.org)). Other entities seeking permission to reproduce this document, in whole or in part, must also obtain permission from the IEEE Standards Activities Department.

IEEE Standards Activities Department

445 Hoes Lane

Piscataway, NJ 08854, USA

Abstract: This standard specifies IEEE 802® media access-independent mechanisms that

optimize handovers between heterogeneous IEEE 802 systems and between IEEE 802 systems

and cellular systems.

Keywords: management, media independent services, MN, mobility, seamless, point of attachment, point of service

[[1]](#footnote-1)•

Notice and Disclaimer of Liability Concerning the Use of IEEE Documents: IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Comittees of the IEEE Standards Association (IEEE-SA) Standards Board. IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

Use of an IEEE Standard is wholly voluntary. IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon any IEEE Standard document.

IEEE does not warrant or represent the accuracy or content of the material contained in its standards, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained in its standards is free from patent infringement. IEEE Standards documents are supplied "AS IS."

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE standard is subjected to review at least every ten years. When a document is more than ten years old and has not undergone a revision process, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE standard.

In publishing and making its standards available, IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing any IEEE Standards document, should rely upon his or her own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

Translations: The IEEE consensus development process involves the review of documents in English only. In the event that an IEEE standard is translated, only the English version published by IEEE should be considered the approved IEEE standard.

Official Statements: A statement, written or oral, that is not processed in accordance with the IEEE-SA Standards Board Operations Manual shall not be considered the official position of IEEE or any of its comittees and shall not be considered to be, nor be relied upon as, a formal position of IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position of IEEE.

Comments on Standards: Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE. However, IEEE does not provide consulting information or advice pertaining to IEEE Standards documents. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important to ensure that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Comittees are not able to provide an instant response to comments or questions except in those cases where the matter has previously been addressed. Any person who would like to participate in evaluating comments or revisions to an IEEE standard is welcome to join the relevant IEEE working group at <http://standards.ieee.org/develop/wg/>.

Comments on standards should be submitted to the following address:

Secretary, IEEE-SA Standards Board

445 Hoes Lane

Piscataway, NJ 08854

USA

Photocopies: Authorization to photocopy portions of any individual standard for internal or personal use is granted by The Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Notice to users

Laws and regulations

Users of IEEE Standards documents should consult all applicable laws and regulations. Compliance with the provisions of any IEEE Standards document does not imply compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

Copyrights

This document is copyrighted by the IEEE. It is made available for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making this document available for use and adoption by public authorities and private users, the IEEE does not waive any rights in copyright to this document.

Updating of IEEE documents

Users of IEEE Standards documents should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect. In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit the IEEE-SA Website at <http://standards.ieee.org/index.html> or contact the IEEE at the address listed previously. For more information about the IEEE Standards Association or the IEEE standards development process, visit IEEE-SA Website at <http://standards.ieee.org/index.html>.

Errata

Errata, if any, for this and all other standards can be accessed at the following URL: <http://standards.ieee.org/findstds/errata/index.html>. Users are encouraged to check this URL for errata periodically.

Patents

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken by the IEEE with respect to the existence or validity of any patent rights in connection therewith. If a patent holder or patent applicant has filed a statement of assurance via an Accepted Letter of Assurance, then the statement is listed on the IEEE-SA Website at <http://standards.ieee.org/about/sasb/patcom/patents.html>. Letters of Assurance may indicate whether the Submitter is willing or unwilling to grant licenses under patent rights without compensation or under reasonable rates, with reasonable terms and conditions that are demonstrably free of any unfair discrimination to applicants desiring to obtain such licenses.

Essential Patent Claims may exist for which a Letter of Assurance has not been received. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims, or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

Participants

At the time this draft standard was completed, the 802.21 Working Group had the following membership:

Subir Das, Chair

, Vice Chair

Participant1

Participant2

Participant3

Participant4

Participant5

Participant6

Participant7

Participant8

Participant9

The following members of the <individual/entity> balloting comittee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

[To be supplied by IEEE]

Balloter1

Balloter2

Balloter3

Balloter4

Balloter5

Balloter6

Balloter7

Balloter8

Balloter9

When the IEEE-SA Standards Board approved this on , it had the following membership:

[To be supplied by IEEE]

<Name>, Chair

<Name>, Vice Chair

<Name>, Past Chair

<Name>, Secretary

SBMember1

SBMember2

SBMember3

SBMember4

SBMember5

SBMember6

SBMember7

SBMember8

SBMember9

\*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

<Name>, DOE Representative

<Name>, NIST Representative

<Name>

IEEE Standards Program Manager, Document Development

<Name>

IEEE Standards Program Manager, Technical Program Development

Introduction

This introduction is not part of P802.21/D, Draft Standard for Local and metropolitan area networks—Part 21: Media Independent Services.

This standard defines extensible media access independent mechanisms that enable the optimization of handovers between heterogeneous IEEE 802 systems and may facilitate handovers between IEEE 802 systems and cellular systems.

Contents

[1. Overview 13](#_Toc382860153)

[1.1 Scope 13](#_Toc382860154)

[1.2 Purpose 13](#_Toc382860155)

[1.3 General 13](#_Toc382860156)

[1.4 Assumptions 15](#_Toc382860157)

[1.5 Media independence 15](#_Toc382860158)

[2. Normative references 17](#_Toc382860159)

[3. Definitions 20](#_Toc382860160)

[4. Abbreviations and acronyms 24](#_Toc382860161)

[5. General architecture 27](#_Toc382860162)

[5.1 Introduction 27](#_Toc382860163)

[5.2 General design principles 27](#_Toc382860164)

[5.3 MISF service overview 27](#_Toc382860165)

[5.4 Media independent service reference framework 31](#_Toc382860166)

[5.5 MISF reference models for link-layer technologies 35](#_Toc382860167)

[5.6 Service access points (SAPs) 42](#_Toc382860168)

[5.7 MIS protocol 44](#_Toc382860169)

[6. MISF services 46](#_Toc382860170)

[6.1 General 46](#_Toc382860171)

[6.2 Service management 46](#_Toc382860172)

[6.3 Media independent event service 47](#_Toc382860173)

[6.4 Media independent command service 53](#_Toc382860174)

[6.5 Media independent information service 57](#_Toc382860175)

[7. Service access points (SAPs) and primitives 71](#_Toc382860176)

[7.1 Introduction 71](#_Toc382860177)

[7.2 SAPs 71](#_Toc382860178)

[7.3 MIS\_LINK\_SAP primitives 74](#_Toc382860179)

[7.4 MIS\_SAP primitives 86](#_Toc382860180)

[7.5 MIS\_NET\_SAP primitives 1](#_Toc382860181)

[8. Media independent services protocol 5](#_Toc382860182)

[8.1 Introduction 5](#_Toc382860183)

[8.2 MIS protocol description 5](#_Toc382860184)

[8.3 MIS protocol identifiers 18](#_Toc382860185)

[8.4 MIS protocol frame format 18](#_Toc382860186)

[8.5 Message parameter TLV encoding 26](#_Toc382860187)

[8.6 MIS protocol messages 27](#_Toc382860188)

[Annex A (informative) Bibliography 80](#_Toc382860189)

[Annex B (normative) Quality of service mapping 82](#_Toc382860190)

[B.1 Generic IEEE 802.21 QoS flow diagram 83](#_Toc382860191)

[B.2 Generic IEEE 802.21 QoS parameter mappings 85](#_Toc382860192)

[B.3 Deriving generic IEEE 802.21 QoS parameters 90](#_Toc382860193)

[Annex C (to be excluded) (informative) Handover procedures 93](#_Toc382860194)

[Annex D (normative) Mapping MIS messages to reference points 94](#_Toc382860195)

[Annex E (normative) Media specific mapping for SAPs 96](#_Toc382860196)

[E.1 MIS\_LINK\_SAP mapping to specific technologies 96](#_Toc382860197)

[E.2 Mappings from MIS\_LINK\_SAP to media-specific SAPs 99](#_Toc382860198)

[Annex F (normative) Data type definition 101](#_Toc382860199)

[F.1 General 101](#_Toc382860200)

[F.2 Basic data types 101](#_Toc382860201)

[F.3 Derived data types 103](#_Toc382860202)

[Annex G (normative) Information element identifiers 138](#_Toc382860203)

[Annex H (normative) MIIS basic schema 140](#_Toc382860204)

[Annex I (informative) Making user extensions to MIIS schema 158](#_Toc382860205)

[Annex J (normative) IEEE 802.21 MIB 159](#_Toc382860206)

[J.1 Parameters requiring MIB definition 159](#_Toc382860207)

[J.2 IEEE 802.21 MIB definition 160](#_Toc382860208)

[Annex K Fragmentation (informative) 172](#_Toc382860209)

[K.1 Example MIS message fragmentation 172](#_Toc382860210)

[K.2 Calculation of securityOverhead when there is an MIS SA 172](#_Toc382860211)

[Annex L (normative) MIS protocol message code assignments 176](#_Toc382860212)

[Annex M (normative) Protocol implementation conformance statement (PICS) proforma 181](#_Toc382860213)

[M.1 Introduction 181](#_Toc382860214)

[M.2 Scope 181](#_Toc382860215)

[M.3 Conformance 181](#_Toc382860216)

[M.4 Instructions 181](#_Toc382860217)

[M.5 Identification of the implementation 184](#_Toc382860218)

[M.6 Identification of the protocol 185](#_Toc382860219)

[M.7 Identification of corrigenda to the protocol 185](#_Toc382860220)

[M.8 PICS proforma tables 185](#_Toc382860221)

[Annex N Authentication and key distribution procedures 191](#_Toc382860222)

[N.1 MIS service access authentication 191](#_Toc382860223)

[N.2 Push key distribution 193](#_Toc382860224)

[N.3 Proactive authentication 194](#_Toc382860225)

[N.4 Optimized pull key distribution 195](#_Toc382860226)

[N.5 Termination phase 196](#_Toc382860227)

[Annex O Protection through transport protocols 198](#_Toc382860228)

[O.1 Protection through layer 2 198](#_Toc382860229)

[O.2 Protection through IPsec 198](#_Toc382860230)

List of Figures

[Figure 1 —MIS services and their initiation 20](#_Toc367177848)

[Figure 2 —MISF communication model 37](#_Toc367177849)

[Figure 3 —Example of network model with MIS services 39](#_Toc367177850)

[Figure 4 —General MISF reference model and SAPs 41](#_Toc367177851)

[Figure 5 —Types of MISF relationship 42](#_Toc367177852)

[Figure 6 —MIS reference model for IEEE 802.3 43](#_Toc367177853)

[Figure 7 —MIS reference model for IEEE 802.11 44](#_Toc367177854)

[Figure 8 —MIS reference model for IEEE 802.16 45](#_Toc367177855)

[Figure 9 —MIS reference model for 3GPP systems 46](#_Toc367177856)

[Figure 10 —MIS reference model for 3GPP2 systems 46](#_Toc367177857)

[Figure 11 —Relationship between different MISF SAPs 48](#_Toc367177858)

[Figure 12 —Link events and MIS events 53](#_Toc367177859)

[Figure 13 —Remote MIS events 54](#_Toc367177860)

[Figure 14 —MIS events subscription and flow 55](#_Toc367177861)

[Figure 15 —Link commands and MIS commands 58](#_Toc367177862)

[Figure 16 —Remote MIS command 59](#_Toc367177863)

[Figure 17 —Command service flow 60](#_Toc367177864)

[Figure 18 —Depicting a list of neighboring networks with information elements 68](#_Toc367177865)

[Figure 19 —TLV representation of information elements 70](#_Toc367177866)

List of Tables

[Table 1 —Summary of reference points 38](#_Toc367177867)

[Table 2 —MIS protocol Ethernet type 50](#_Toc367177868)

[Table 3 —Service management primitives 51](#_Toc367177869)

[Table 4 —Link events 56](#_Toc367177870)

[Table 5 —MIS events 57](#_Toc367177871)

[Table 6 —Link commands 61](#_Toc367177872)

[Table 7 —MIS commands 61](#_Toc367177873)

[Table 8 —Naming convention for MIS handover command primitives 62](#_Toc367177874)

[Table 9 —Information element containers 65](#_Toc367177875)

[Table 10 —Information elements 65](#_Toc367177876)

[Table 11 —Information element namespace 69](#_Toc367177877)

[Table 12 —IE\_CONTAINER\_LIST\_OF\_NETWORKS definition 70](#_Toc367177878)

[Table 13 —IE\_CONTAIN ER\_NETWORK definition 71](#_Toc367177879)

[Table 14 —IE\_CONTAINER\_POA definition 71](#_Toc367177880)

[Table 15 —MIS\_LINK\_SAP primitives 75](#_Toc367177881)

[Table 16 —MIS\_NET\_SAP primitive 76](#_Toc367177882)

[Table 17 —MIS\_SAP primitives 76](#_Toc367177883)

Draft Standard for   
Local and metropolitan area networks  
  
Part 21: Media Independent Services

IMPORTANT NOTICE: IEEE Standards documents are not intended to ensure safety, health, or environmental protection, or ensure against interference with or from other devices or networks. Implementers of IEEE Standards documents are responsible for determining and complying with all appropriate safety, security, environmental, health, and interference protection practices and all applicable laws and regulations.

This IEEE document is made available for use subject to important notices and legal disclaimers.   
These notices and disclaimers appear in all publications containing this document and may   
be found under the heading “Important Notice” or “Important Notices and Disclaimers   
Concerning IEEE Documents.” They can also be obtained on request from IEEE or viewed at <http://standards.ieee.org/IPR/disclaimers.html>.

1. Overview
   1. Scope

This standard defines extensible IEEE 802® media access independent mechanisms that enable the optimization of handover between heterogeneous IEEE 802 networks and facilitates handover between IEEE 802 networks and cellular networks.

* 1. Purpose

The purpose is to improve the user experience of mobile devices by facilitating handover between IEEE 802 networks whether or not they are of different media types, including both wired and wireless, where handover is not otherwise defined; and to make it possible for mobile devices to perform seamless handover where the network environment supports it. These mechanisms are also usable for handovers between IEEE 802 networks and non IEEE 802 networks.

* 1. General

This standard provides link-layer intelligence and other related network information to upper layers to optimize handovers between heterogeneous networks. This includes media types specified by Third Generation (3G) Partnership Project (3GPP), 3G Partnership Project 2 (3GPP2), both wired and wireless media in the IEEE 802 family of standards, and downlink-only (DO) media such as Digital Video Broadcasting (DVB), Terrestrial Digital Multimedia Broadcasting (T-DMB) and Advanced Television Systems Committee–Mobile/Handheld (ATSC-M/H). In this standard, unless otherwise noted, *media* refers to the method/mode of accessing a telecommunication system (e.g., cable, radio, satellite), as opposed to sensory aspects of communication (e.g., audio, video).

The following items are not within the scope of this standard:

* Intra-technology handover [except for handovers across extended service sets (ESSs) in case of IEEE 802.11]
* Handover policy
* Security mechanisms
* Enhancements specific to particular link-layer technologies that are required to support this standard (they will be carried out by those respective link-layer technology standards)
* Higher layer (layer 3 and above) enhancements that are required to support this standard

The purpose of this standard is to enhance the experience of mobile users by facilitating handovers between heterogeneous networks. The standard addresses the support of handovers for both mobile and stationary users. For mobile users, handovers can occur when wireless link conditions change due to the users' movement. For the stationary user, handovers become imminent when the surrounding network environment changes, making one network more attractive than another.

This standard supports another important aspect of optimized handover—link adaptation. A user can choose an application that requires a higher data rate than available on the current link, necessitating a link adaptation to provide the higher rate, or necessitating a handover if the higher rate is unavailable on the current link.

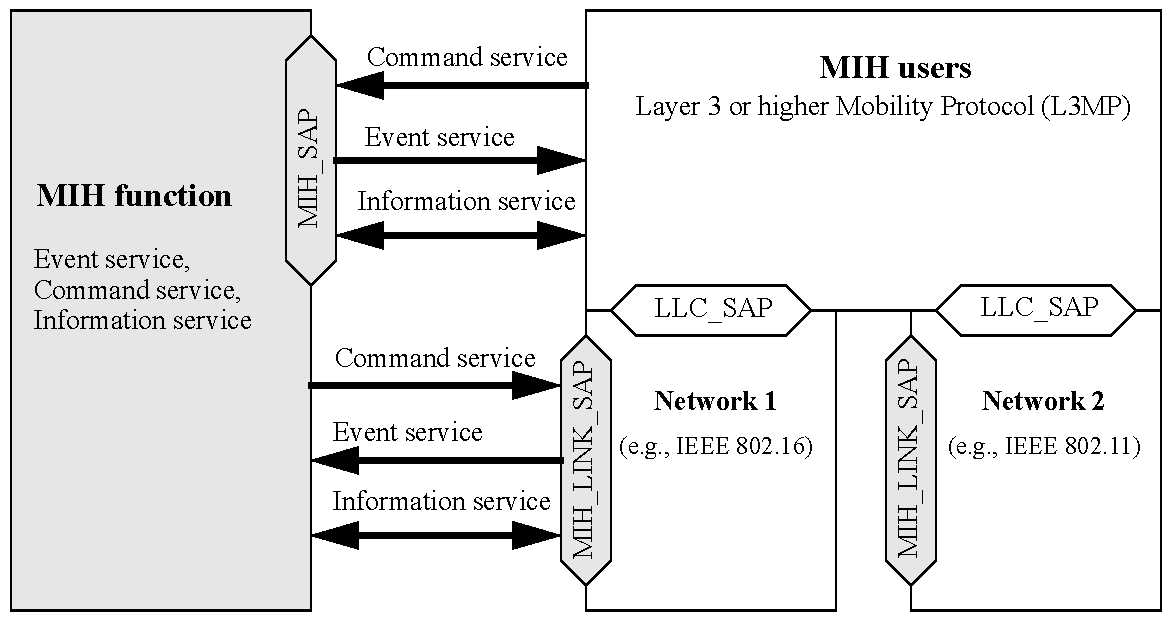
In all such cases, service continuity should be maintained to the extent possible during handover. As an example, when making a network transition during a phone call the handover procedures should be executed in such a way that any perceptible interruption to the conversation will be minimized.

This standard supports cooperative use of information available at the MN and within the network infrastructure. The MN is well-placed to detect available networks. The network infrastructure is well-suited to store overall network information, such as neighborhood cell lists, location of MNs, and higher layer service availability. Both the MN and the network make decisions about connectivity. In general, both the MN and the network points of attachment (such as base stations and access points) can be multi-modal (i.e., capable of supporting multiple radio standards and simultaneously supporting connections on more than one radio interface).

The overall network can include a mixture of cells of drastically different sizes, such as those from IEEE 802.15, IEEE 802.11, IEEE 802.16, 3GPP, and 3GPP2, with overlapping coverage. The handover process can be initiated by measurement reports and triggers supplied by the link layers on the MN. The measurement reports can include metrics such as signal quality, synchronization time differences, and transmission error rates. Specifically the standard consists of the following elements:

1. A framework that enables service continuity while a MN transitions between heterogeneous link-layer technologies. The framework relies on the presence of a mobility management protocol stack within the network elements that support the handover. The framework presents media independent service (MIS) reference models for different link-layer technologies.
2. A set of handover-enabling functions within the protocol stacks of the network elements and a new entity created therein called the MIS Function (MISF).
3. A media independent service access point (called the MIS\_SAP) and associated primitives are defined to provide MIS users with access to the services of the MISF. The MISF provides the following services:
   1. The media independent event service that detects changes in link-layer properties and initiates appropriate events (triggers) from both local and remote interfaces.
   2. The media independent command service provides a set of commands for the MIS users to control link properties that are relevant to handover and switch between links if required.
   3. The media independent information service provides the information about different networks and their services thus enabling more effective handover decision to be made across heterogeneous networks.
4. The definition of new link-layer service access points (SAPs) and associated primitives for each link-layer technology. The new primitives help the MISF collect link information and control link behavior during handovers. If applicable, the new SAPs are recommended as amendments to the standards for the respective link-layer technology.

Figure 1 shows the placement of the MISF within the protocol stack of a multiple interfaced MN or network entity. The MISF provides services to the MIS users through a single media independent interface (the MIS service access point) and obtains services from the lower layers through a variety of media dependent interfaces (media-specific SAPs). <ISSUE: change figure to say “Service Protocol” instead> <ISSUE: Change MIH to MIS or MIC in figure>



1. —MIS services and their initiation
   1. Assumptions

The following assumptions have been made in the development of this standard:

1. The MN is capable of supporting multiple link-layer technologies, such as wireless, wired, or mixed.
2. The MISF is a logical entity, whose definition is independent of its deployment location on the MN or in the network.
3. The MISF, regardless of whether it is located on the MN or in the network, receives and transmits information about the configuration and condition of access networks around the MN. This information originates at different layers of the protocol stack within the MN or at various network elements.
   1. When the information originates at a remote network element, the MISF on the local network element obtains it through MIS message exchanges with a peer MISF instance that resides in the remote network element.
   2. When the information originates at lower layers of the protocol stack within an MN or network entity, the MISF on that entity obtains it locally through the service primitives of the SAPs that define the interface of the MISF with the lower layers.
   3. Media independence

The intent of this standard is to provide generic link-layer intelligence independent of the specifics of MNs or radio networks. As such, this standard is intended to provide a generic interface between the mobility-management protocol stack and existing media-specific link layers, such as those specified by 3GPP, 3GPP2, the IEEE 802 family of standards, and downlink-only media.

This standard defines SAPs and primitives that provide generic link-layer intelligence. Individual media- specific technologies thereafter need to enhance their media-specific SAPs and primitives to satisfy the generic abstractions of this standard. Suitable amendments are required to existing link-layer [medium access control (MAC)/ physical layer (PHY)] standards of different media-specific technologies such as IEEE Std 802.3™, IEEE Std 802.1 1™, IEEE Std 802.16™, 3GPP, 3GPP2, and DVB to satisfy the requirements of generic link-layer intelligence identified by this standard.[[2]](#footnote-2)

1. Normative references

The following referenced documents are indispensable for the application of this document (i.e., they must be understood and used, so each referenced document is cited in text and its relationship to this document is explained). For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

3GPP TS 23.003 (2007-09), Numbering, addressing and identification (Release 7).

3GPP TS 25.008, Digital cellular telecommunication system (Phase 2+); Radio subsystem link control.

3GPP TS 25.2 15 (2007-11), Physical layer - Measurements (FDD) (Release 7).

3GPP TS 25.401 (2007-09), UTRAN overall description (Release 7).

3GPP TS 25.413 (2007-09), UTRAN Iu interface RANAP signalling (Release 7).

3GPP2 C.S0004-D (2004-02), Signaling Link Access Control (LAC) Standard for cdma2000 Spread Spectrum Systems.

ANSI X3.159-1989: Programming Language C.[[3]](#footnote-3)

IEEE P802.1aj™/D2.2, (2007-10), Draft Standard for Local and Metropolitan Area Networks—Virtual Bridged Local Area Networks—Amendment 08: Two-Port Media Access Control (MAC) Relay.[[4]](#footnote-4), [[5]](#footnote-5), [[6]](#footnote-6)

IEEE P802.1 1u™/D3.0, (2008-05) (this version), Draft Amendment to Standard for Information Technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements; Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 7: Interworking with External Networks.

IEEE P802.1 6Rev2/D5.0, (2008-06), Information Technology—Telecommunications and information exchange between system-Local and metropolitan area networks—Specific Requirements—Part 16: Air Interface for Fixed Broadband Wireless Access Systems.

IEEE Std 802.1 1™-2007, Information Technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.

IEEE Std 802.1 1k™-2008, Information Technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements; Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications; Amendment 1: Radio Resource Measurement of Wireless LANs.

IEEE Std 802.16™-2004 [ISO/IEC 8802-16: 2004], Information Technology—Telecommunications and information exchange between system—Local and metropolitan area networks—Specific Requirements—Part 16: Air Interface for Fixed Broadband Wireless Access Systems.

IEEE Std 802.16e™-2005 [ISO/IEC 8802-16: 2005], Information Technology—Telecommunications and information exchange between system—Local and metropolitan area networks—Specific Requirements—Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1.

IEEE Std 802.3™-2005, Information Technology—Telecommunications and information exchange between system—Local and metropolitan area networks—Specific Requirements—Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications.

IETF RFC 1661 (1994-07), The Point-to-Point Protocol (PPP).[[7]](#footnote-7)

IETF RFC 2865 (2000-06), Remote Authentication Dial In User Service (RADIUS). IETF RFC 2988 (2000-11), Computing TCP's Retransmission Timer.

IETF RFC 3344 (2002-08), IP Mobility Support for IPv4.

IETF RFC 3588 (2003-09), Diameter Base Protocol.

IETF RFC 3775 (2004-06), Mobility Support in IPv6.

IETF RFC 3825 (2004-07), Dynamic Host Configuration Protocol Option for Coordinate-based Location Configuration Information.

IETF RFC 4119 (2005-12), A Presence-based GEOPRIV Location Object Format.

IETF RFC 4140 (2005-08), Hierarchical Mobile IPv6 Mobility Management (HMIPv6).

IETF RFC 4282 (2005-12), The Network Access Identifier.

IETF RFC 4555 (2006-06), IKEv2 Mobility and Multihoming Protocol (MOBIKE).

IETF RFC 4776 (2006-11), Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Option for Civic Addresses Configuration Information.

IETF RFC 4857 (2007-06), Mobile IPv4 Regional Registration.

IETF RFC 4881 (2007-06), Low-Latency Handoffs in Mobile IPv4.

IETF RFC 5268 (2008-06), Mobile IPv6 Fast Handovers.

ISO 3166-1 (1997), Codes for the representation of names of countries and their subdivisions—Part 1: Country codes.[[8]](#footnote-8)

ISO 4217, Codes for the Representation of Names of Countries.

ITU-T Recommendation X.290 (1995), OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications—General concepts.[[9]](#footnote-9)

ITU-T Recommendation X.296 (1995), OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications—Implementation conformance statements.

ITU-T Recommendation Y. 1540, Internet protocol data communication service—IP packet transfer and availability performance parameters.

W3C Recommendation, RDF/XML Syntax Specification.[[10]](#footnote-10)

W3C Recommendation, Resource Description Framework (RDF)—Concepts and Abstract Syntax.

W3C Recommendation, SPARQL Query Language for RDF.

1. Definitions

For the purposes of this document, the following terms and definitions apply. The IEEE Standards Dictionary Online should be consulted for terms not defined in this clause. [[11]](#footnote-11)

**authenticated encryption**: An algorithm to convert plaintext data to ciphertext and generate a message authentication code with a cryptographic key as a parameter to provide confidentiality, integrity, and authen- ticity of the data. See also: encryption; MIC algorithm.

**authentication process**: A process to assure that the claimed identity belongs to the entity. It is also called entity authentication. In this standard, an access authentication is an entity authentication with the identity used to access a specific network or a media independent service (MIS).

**authentication server**: A server used for authentication purposes. When EAP is used as an authentication protocol, the authentication server is an EAP server.

**authenticator**: A network entity to execute EAP with a MN called a peer. An authenticator can use a backend server to conduct EAP execution. Syn: EAP authenticator.

**bidirectional network**: A general communication network providing bidirectional transmission such as 802.3, 802.11, 802.16, 3GPP and 3GPP2.

**candidate authenticator**: An authenticator that is associated with a candidate PoA. candidate network: A network that is a potential target to the MN's movement candidate PoS: A potential PoS that can serve the MNs after movement.

**candidate point of attachment (candidate PoA)**: A point of attachment (PoA) under evaluation to which the link may be switched.

**decryption**: An algorithm to convert ciphertext of data to plaintext with a cryptographic key as a parameter. It is an inverse operation of encryption.

**downlink-only (DO) network**: A broadcasting network providing unidirectional transmission from the PoA to the user device, such as DVB, T-DMB and ATSC-M/H.

**dual-radio operation**: In this mode a dual radio device can receive and transmit simultaneously on both the radios. See also: single-radio operation.

**EAP authenticator**: See: authenticator.

**EAP peer**: The entity that responds to the EAP authenticator.

**EAP Re-authentication**: An authentication protocol using a key established in a previous EAP execution as defined in IETF RFC 5296.5

**EAP Server**: The entity that terminates the EAP execution with the EAP peer. In the case where no back- end authentication server is used, the EAP server is a part of the EAP authenticator. In the case where a backend authentication server is used, the EAP server is located on the backend authentication server.

**encryption**: An algorithm to convert plaintext data to ciphertext to provide confidentiality with a crypto- graphic key as a parameter.

**extensible authentication protocol (EAP)**: An access authentication framework specified in IETF RFC 3748. It can support different authentication methods, called EAP methods.

**home subscriber network**: Network managed by an operator with whom the subscriber has a business relationship (subscription). See also: visited network; serving network.

**link**: A communication channel through which nodes communicate for the exchange of L2 protocol data units. Each link is associated with two endpoints and has a unique identifier.

**link indication**: Link state information provided by the link layer to higher layers.

**link layer**: Conceptual layer of control or processing logic that is responsible for maintaining control of the data link. The data link-layer functions provide an interface between the higher-layer logic and the data link.

**link switch**: The process by which a MN changes the link that connects it to the network. Changing a link implies changing the remote link endpoint and therefore the point of attachment of the MN.

**lower layers**: The layers located at OSI Level 2 and below across different link-layer technology standards supported by this standard. For example, the IEEE 802.11 Lower Layers are the MAC sublayer and the PHY, while the 3GPP Lower Layers are L1/MAC/radio link control (RLC)/packet data convergence protocol (PDCP) in the case of wideband code division multiple access (W-CDMA) frequency division duplex (FDD)/time division duplex (TDD), L1/LAPDm in the case of GSM CS, and L1/MAC/RLC in the case of general packet radio service (GPRS)/ Enhanced GPRS (EGPRS), respectively. The term “Lower Layers” also includes Logical Link Control Layers such as IEEE 802.2 Logical Link Control (LLC) or 3GPP Radio Link Control (RLC). The MISF uses the services provided by these layers.

**media independent service (MIS) discovery protocol**: A protocol for discovering media independent service (MIS) entities.

**media independent service (MIS) network entity**: Network entity with at least one media independent service function (MISF).

**media independent service (MIS) node**: An entity providing a MISF (MN or network).

**media independent service (MIS) non-PoS**: An MIS network entity that can directly exchange MIS messages with other MIS network entities but cannot directly exchange MIS messages with any MIS enabled MN.

**media independent service (MIS) transport protocol**: A protocol for transporting MIS protocol messages between a pair of MIS entities.

**media independent service (MIS) users**: Entities that use the services provided by the MISF. MIS users use the MIS\_SAP to interact with the MISF.

**media independent service function (MISF)**: A function that realizes at least one media independent service.

**media independent service function (MISF) pairing**: The communication relationship that exists between different MISF instances when they exchange MIS messages.

**media independent service function (MISF) transaction**: A combination of an MIS Request message and MIS Response message, MIS Indication, or MIS Response message and any associated MIS Acknowledgement messages.

**media independent point of service (MIS PoS)**: Network-side MISF instance that exchanges MIS messages with an MN-based MISF. The same MIS Network Entity includes an MIS PoS for each MIS-enabled MN with which it exchanges MIS messages. A single MIS PoS can host more than one MIS service. The same MIS Network Entity can include multiple MIS Points of Service that can provide different combinations of MIS services to the respective MNs based on subscription or roaming conditions. Note that for a network entity comprising multiple interfaces, the notion of MIS PoS is associated with the network entity itself and not with just one of its interfaces. For MIS service access authentication, a PoS serves as an authenticator. Moreover, when a service access authentication establishes keys for proactive authentication, a PoS provides key distribution service for media specific authenticators.

**media specific authentication server**: An authentication server used for media specific access authentication.

**media specific authenticator**: An authenticator used for a media specific network access authentication.

**media specific network access authentication**: An authentication protocol for media access purpose specified for a specific media access. It may establish keys to be used in media specific protection mechanisms.

**media specific protection mechanism**: A mechanism that is applied to media specific layers to protect the data traffic using an encryption algorithm, an integrity protection algorithm, an authenticated encryption algorithm, or a combination of an encryption algorithm and an integrity protection algorithm.

**message authentication code (a.k.a. message integrity code)**: A data string generated over a message with a symmetric key by an algorithm, called message authentication code algorithm. It is used to verify the integrity of the message and to authenticate the origin of the message.

**message authentication code algorithm**: An algorithm to generate a message authentication code on a data message with a symmetric key to provide integrity protection and message origination authentication. See: message authentication code.

**message integrity code (MIC)**: See: message authentication code.

**MIS security association (SA)**: An MIS security association is a set of cryptographic attributes established between the peer MIS entities for protecting MIS messages at the MIS protocol layer. An MIS SA is established via TLS handshake or EAP execution, where both the TLS handshake and EAP execution take place over the MIS protocol. When an MIS SA is established via TLS handshake, the TLS master key and its child keys, TLS random values and the TLS cipher suite negotiated in the TLS handshake are a part of the MIS SA. When an MIS SA is established via EAP execution, an MSK or rMSK and its child keys, MIS random values and the MIS cipher suite negotiated between the peer MIS entities are associated with the MIS SA.

**MIS service access authentication**: An authentication process that authorizes the access to media indepen- dent services.

**MIS service access authentication server**: An authentication server used to execute the MIS service access authentication. See: authentication server.

**mobile node (MN)**: Communication node that can change its point of attachment from one link to another.

**multimedia program (MMP)**: An instance of certain content (e.g., voice, data or video) with some specific attributes, e.g., chapter 2 of a TV series.

**multimedia service (MMS)**: A sequence of MMPs under the control of a content aggregator and provider, e.g., TV Channel One, TV Channel Two, etc.

**network detection**: The process by which a MN collects information on networks in its locality, identifies the different points of attachment, and ascertains the validity of link-layer configuration.

**network entity**: A communication node inside the network.

**network neighborhood**: The area of interest in which the network discovery and selection entity seeks to determine the available coverage of a wired/wireless network with identical or different link-layer technologies.

**network point of attachment (network PoA, or PoA)**: The network side endpoint of a layer 2 link that includes a MN as the other endpoint. See also: candidate PoA; serving PoA; target PoA.

**network selection**: The process by which a MN or a network entity makes a decision to connect to a specific network (possibly out of many available) based on a policy configured in the MN and/or obtained from the network.

**network selector**: The entity that undertakes the network selection decisions.

**operator identifier (operator ID)**: An identifier of the access or core network provider.

**PICS Proforma**: A normative document to express in compact form the static conformance requirements of a specification. As such, it serves as a reference to the static conformance review.

**proactive authentication**: A media specific authentication with a candidate network(s).

**protection mechanisms** for MIS messages: A protection mechanism that is applied to MIS PDU using an encryption algorithm, an integrity protection algorithm, an authenticated encryption algorithm, or a combi- nation of an encryption algorithm and an integrity protection algorithm.

**security association identifier (SAID)**: An identifier of an MIS security association. When an SA is estab- lished through TLS, it is the TLS session ID. When an SA is generated through an EAP execution, it is assigned by the authenticator and the ID value is an octet string unique for a pair of MIS functions.

**serving authenticator**: The authenticator which is associated with the serving PoA.

**serving network**: A network that provides services to the user. The serving network can be a home subscriber network or a visited network. See also: visited network; home subscriber network.

**serving point of attachment (serving PoA)**: The PoA of the current link being used by the MN.

**serving PoS**: An MIS PoS that is currently providing a media independent service to the MN.

**single-radio operation**: In this mode, a dual radio device can receive and transmit on only one radio at a time. This is usually the mode of operation when radio frequencies of the two radios are close to each other (e.g., in IMT 2000 bands). See also: dual-radio operation.

**static conformance requirement**: One of the requirements that specify the limitations on the combinations of implemented capabilities permitted in a real open system, which is claimed to conform to the relevant specification(s).

**static conformance review**: A review of the extent to which the static conformance requirements are claimed to be supported by the system under test, by comparing the answers in the implementation conformance statement(s) and the system conformance statement with the static conformance requirements expressed in the relevant specifications.

**target point of attachment (target PoA)**: A candidate PoA that has been selected to become the new serving PoA.

**uniform resource identifier (URI)**: A compact sequence of characters that identifies an abstract or physical resource including video.

**visited network**: A network managed by an operator other than the subscriber’s home operator and in which the subscriber is receiving service. See also: home subscriber network; serving network.

1. Abbreviations and acronyms

The following abbreviations and acronyms are used in this standard:

|  |  |
| --- | --- |
| 3G | 3rd generation |
| 3GPP | 3rd Generation Partnership Project |
| 3GPP2 | 3rd Generation Partnership Project 2 |
| AAA | authentication, authorization, and accounting |
| ACK | acknowledgement |
| AES | advanced encryption standard |
| AID | action identifier |
| AP | access point |
| AR | access router |
| AS | authentication server |
| BCE | binding cache entry |
| BS | base station |
| BTS | base transceiver station |
| CBC | cipher block chaining |
| CCM | counter with CBC message authentication code |
| CoA | care-of address |
| CoS | class of service |
| CS | convergence sublayer / command service |
| DCD | downlink channel descriptor |
| DHCP | dynamic host configuration protocol |
| DTLS | datagram transport layer security |
| EAP | extensible authentication protocol |
| ERP | EAP Re-authentication Protocol |
| ES | event service |
| ESS | extended service set |
| FA | foreign agent |
| GPRS | general packet radio service |
| GSM | global system for mobile communication |
| HESSID | homogenous extended service set ID |
| HMAC | keyed-hash message authentication code |
| IEEE | Institute of Electrical and Electronics Engineers |
| IETF | Internet Engineering Task Force |
| IP | internet protocol |
| IS | information service |
| ITU | International Telecommunications Union |
| IV | Initialization vector |
| L1 | layer 1 (PHY) |
| L2 | layer 2 (MAC and/or LLC) |
| LAN | local area network |
| LbyR | location by reference |
| LCP | location configuration protocol |
| LLC | logical link control |
| LMA | local mobility anchor |
| LSAP | logical link control service access point |
| LTE | long term evolution |
| MAC | medium access control |
| MAG | mobile access gateway |
| MIAK | media independent authentication key |
| MIC | message integrity code |
| MICS | media independent command services |
| MIEK | media independent encryption key |
| MIES | media independent event services |
| MIIK | media independent integrity key |
| MIIS | media independent information service |
| MISK | media independent session key |
| MLME | MAC layer management entity |
| MN | mobile node |
| MPLS | multi-protocol label switching |
| MSA | media specific authenticator |
| MSB | most significant bit |
| MSDU | medium access control (MAC) service data unit |
| MSGCF | MAC state generic convergence function |
| MSK | master session key |
| MSPMK | media specific pairwise master key |
| MSRK | media specific root key |
| N/A | not applicable |
| NAI | network access identifier |
| NAS | network access server |
| NCMS | network control and management system |
| OUI | organizationally unique identifier |
| PDU | protocol data unit |
| PHY | physical layer |
| PLME | physical layer management entity |
| PLMN | public land mobile network |
| PoA | point of attachment |
| PoS | point of service |
| PPP | point-to-point protocol |
| PRF | pseudorandom function |
| PSAP | public safety answering point |
| QoS | quality of service |
| RAT | radio access technology |
| RDF | resource description framework |
| RFC | request for comment |
| RLC | radio link control |
| rMSK | re-authentication master session key |
| RNC | radio network controller |
| RSNA | robust security network association |
| RSSI | received signal strength indication |
| SA | security association |
| SAID | security association identifier |
| SAP | service access point |
| SCTP | stream control transmission protocol |
| SDO | standards development organization |
| SDU | service data unit |
| SHA | secure hash algorithm |
| SIB | system information block |
| SID | service identifier |
| SINR | signal over interference plus noise ratio |
| SIP | session initiation protocol |
| SLA | service level agreement |
| SM | session management |
| SME | station management entity |
| SNR | signal-to-noise ratio |
| SS | subscriber station |
| STA | station |
| TCP | transmission control protocol |
| TLS | transport layer security |
| TLV | type-length-value |
| UDP | user datagram protocol |
| UE | user equipment |
| UIR | unauthenticated information request |
| UMTS | universal mobile telecommunications system |
| URL | uniform resource locator |
| WLAN | wireless local area network |
| XML | extensible mark-up language |

1. General architecture
   1. Introduction

In the base REVP document, the subsubsections of this subsection was deleted (including 5.1.9 from 802.21a), since they focused on specific applications. A new “Introduction” is needed, but there is some weird template requirement for heavy indentation that needs to be revisited.

* 1. General design principles

MISF design principles

This standard is based on the following general design principles:

1. MISF is a logical entity that facilitates handover decision making. MIS users make handover decisions based on inputs from the MISF.
2. MISF provides abstracted services to higher layers. The service primitives defined by this interface are based on the technology-specific protocol entities of the different access networks. The MISF communicates with the lower layers of the mobility-management protocol stack through technology-specific interfaces.
3. Higher layer mobility management protocols specify handover signaling mechanisms for vertical handovers. Additionally, different access network technologies have defined handover signaling mechanisms to facilitate horizontal handover. The definition of such handover signaling mechanisms is outside the scope of this standard except in the case of handovers across ESSs in IEEE Std 802.11. The role of this standard (IEEE Std 802.21) is to serve as a handover facilitating service and to maximize the efficiency of such handovers by providing appropriate link-layer intelligence and network information.
4. The standard provides support for remote events. Events are advisory in nature. The decision whether to cause a handover or not based on these events is outside the scope of this standard.
5. The standard supports transparent operation with legacy equipment. IEEE 802.21 standard compatible equipment should be able to co-exist with legacy equipment.

QoS design principles

In the context of this standard it is assumed that applications communicate via a communication channel that is considered to be composed of several connected segments, each under a possibly different but cooperative administrative authority. Examples of such channels [e.g., for internet protocol (IP) traffic] have been detailed in ITU-T Recommendation Y. 1540.

It is generally accepted that, based on the required accuracy of information transfer, applications can be grouped into a small number of behavioral sets (ITU-T Recommendation Y. 1540) called *class of service* (CoS). Support for differentiation via CoS is pervasive in many of the IEEE 802 based standards (IEEE Std 802.11, IEEE Std 802.1qTM, IEEE Std 802.16, etc.).

It is assumed that the classes of service definitions used within this standard conform to ITU-T Recommendation Y. 1540.

* 1. MISF service overview

General

This standard defines services that comprise the MISF service; these services facilitate handovers between heterogeneous access links.

1. A media independent event service (MIES) that provides event classification, event filtering and event reporting corresponding to dynamic changes in link characteristics, link status, and link quality.
2. A media independent command service (MICS) that enables MIS users to manage and control link behavior relevant to handovers and mobility.
3. A media independent information service (MIIS) that provides details on the characteristics and services provided by the serving and neighboring networks. The information enables effective system access and effective handover decisions.

The MISF provides asynchronous and synchronous services through well-defined SAPs for link layers and MIS users. In the case of a system with multiple network interfaces of arbitrary type, the MIS users use the event service, command service, and information service provided by MISF to manage, determine, and control the state of the underlying interfaces.

These services provided by MISF help the MIS users in maintaining service continuity, service adaptation to varying quality of service, battery life conservation, network discovery, and link selection. In a system containing heterogeneous network interfaces of IEEE 802 types and cellular (3GPP, 3GPP2) types, the MISF helps the MIS users to implement effective procedures to couple services across heterogeneous network interfaces. MIS users utilize services provided by the MISF across different entities to query resources required for a handover operation between heterogeneous networks.

MIS Services in MNs facilitate seamless handovers between heterogeneous networks. MIS Services are used by MIS users such as a mobility management protocol (e.g., Mobile IP). Other mobility management protocols (in addition to Mobile IP) and even other MIS users are not precluded from making use of MIS Services.

Media independent event service

* + - * 1. General

Events indicate changes in state and transmission behavior of the physical, data link and logical link layers, or predict state changes of these layers. The event service is also used to indicate management actions or command status on the part of the network or some management entity.

* + - * 1. Event origination

Events originate from the MISF (MIS Events) or any lower layer (Link Events) within the protocol stack of an MN or network node, as shown in Figure 12.

* + - * 1. Event destination

The destination of an event is the MISF or any upper layer entity. The recipient of the event is located within the node that originated the event or within a remote node. The destination of an event is established with a subscription mechanism that enables an MN or network node to subscribe its interest in particular event types.

* + - * 1. Event service flow

In the case of local events, messages often propagate from the lower layers (e.g., PHY, MAC) to the MISF and from MISF to any upper layer. In case of remote events, messages propagate from the MISF in one protocol stack to the MISF in the peer protocol stack. One of the protocol stacks can be present in an MN while the other can be present in a fixed network entity. This network entity is the point of attachment or any node not directly connected to the other protocol stack.

* + - * 1. Event service use cases and functions

The event service is used to detect the need for handovers. For example, an indication that the link will cease to carry MAC service data units (SDUs) at some point in the near future is used by MIS users to prepare a new point of attachment ahead of the current point of attachment ceasing to carry frames. This has the potential to reduce the time needed to handover between attachment points.

Events carry additional context data such as a layer 2 (MAC and/or LLC) (L2) identifier or L3 identifier. A Link\_Up event can also carry a new IP address acquisition indication that informs the upper layers of the need to initiate a layer 3 handover.

Media independent command service

* + - * 1. General

The command service enables higher layers to control the physical, data link, and logical link layers (also known as “lower layers”). The higher layers control the reconfiguration or selection of an appropriate link through a set of handover commands. If an MISF supports the command service, all MIS commands are mandatory in nature. When an MISF receives a command, it is always expected to execute the command.

* + - * 1. Command origination

Commands are invoked by MIS users (MIS Commands), as well as by the MISF itself (Link Commands), as shown in Figure 15.

* + - * 1. Command destination

The destination of a command is the MISF or any lower layer. The recipient of a command is located within the protocol stack that originated the command, or within a remote protocol stack.

* + - * 1. Command service flow

In the case of local commands, messages often propagate from the MIS users (e.g., policy engine) to the MISF and then from MISF to lower layers. In the case of remote commands, messages propagate from MIS users via MISF in one protocol stack to the MISF in a peer protocol stack (with the use of the MIS Protocol). One of the protocol stacks can be present in an MN while the other can be present in a fixed network entity. This network entity is either a point of attachment or any node not directly connected to the other protocol stack.

* + - * 1. Command service use cases and functions

The commands generally carry the upper layer decisions to the lower layers on the local device entity or at  
the remote entity. For example the command service can be used by the policy engine of an entity in the  
network to request an MN to switch between links (remote command to lower layers on MN protocol stack).

This standard facilitates both mobile-initiated and network-initiated handovers. Handovers are initiated by changes in the wireless environment that leads to the selection of a network that supports a different access technology other than the serving network.

During network selection, the MN and the network need to exchange information about available candidate networks and select the best network. The network selection policy engine can select a different network than the current one, which can necessitate an inter-technology handover. Network selection and handover initiation are outside the scope of mobility management protocols such as mobile IP (MIP) and session initiation protocol (SIP). Once a new network has been selected and handover has been initiated, mobility management protocols handle packet routing aspects such as address update and transfer of packet delivery to the new network.

This standard supports a set of media independent commands that help with network selection under different conditions. These commands allow both the MN and the network to initiate handovers and exchange information about available networks and negotiate the best available network under different conditions. Please refer to the flow diagrams in Annex C for more information. These commands do not affect packet routing aspects and can be used in conjunction with other mobility management protocols such as MIP and SIP to perform inter-technology handovers.

Media independent information service

The media independent information service (MIIS) provides a framework and corresponding mechanisms by which an MISF entity can discover and obtain network information existing within a geographical area to facilitate the handovers.

The neighboring network information discovered and obtained by this framework and mechanisms can also be used in conjunction with user and network operator policies for optimum initial network selection and access (attachment), or network re-selection in idle mode.

MIIS primarily provides a set of information elements (IEs), the information structure and its representation, and a query/response type of mechanism (pull mode) for information transfer. The information can also include inter-technology handover policies. The definition of such policies is outside the scope of this standard. MIIS also supports a push mode wherein the information can be pushed to the MN by the operator. The information can be present in an information server from where the MISF in the MN accesses it. The definition of the information server is outside the scope of this standard. In other cases information can be present locally in the MN, and can be learned by the MN or pre-provisioned, or both. The definition of and indexing of such a local database, as well as the regime for maintaining it or accessing it, are outside the scope of this standard.

The information is made available via both lower and higher layers. Information is made available at L2 through both a secure and a non-secure port. Information available through the non-secure port allows a network selection decision to be made before incurring the overhead of authentication and the establishment of a secure L2 connection with the network.

In certain scenarios information cannot be accessed at L2, or the information available at L2 is not sufficient to make an intelligent handover decision. In such cases information can be accessed via higher layers. Hence this standard enables both L2 and L3 transport options for information access. The selected transport option is expected to provide security, such as data integrity and data confidentiality, for the information access.

MIIS typically provides static link-layer parameters such as channel information, the MAC address and security information of a point of attachment (PoA). Information about available higher layer services in a network can also help in more effective handover decision making before the MN actually attaches to any particular network.

The information provided by MIIS conforms to the structure and semantics specified within this standard. MIIS specifies a common (or media independent) way of representing this information across different technologies by using a standardized format such as extensible mark-up language (XML) or binary encoding. A structure of information is defined as a schema.

MIIS provides the ability to access information about all networks in a geographical area from any single L2 network, depending on how the IEEE 802.21 MIIS service is implemented. MIIS either relies on existing access media specific transports and security mechanisms or L3 transport and L3 security mechanisms to provide access to the information. How this information is developed and deployed in a given network is outside the scope of the standard. Typically, in a heterogeneous network composed of multiple media types, the network selector or higher layer mobility management will collect information from different media types and assemble a consolidated view to facilitate its inter-media handover decision.

Some networks such as the cellular networks already have an existing means of detecting a list of neighborhood base stations within the vicinity of an area via the broadcast control channel. Some IEEE standards define similar means and support MNs in detecting a list of neighborhood access points within the vicinity of an area via either beaconing or via the broadcast of MAC management messages. MIIS defines a unified mechanism to the higher layer entities to provide handover candidate information in a heterogeneous network environment by a given geographical location. However, the algorithm for deciding what information to provide is out of scope. In the larger view, the objective is to help the higher layer mobility protocol to acquire a global view of the heterogeneous networks to effect seamless handover across these networks.

* 1. Media independent service reference framework

General

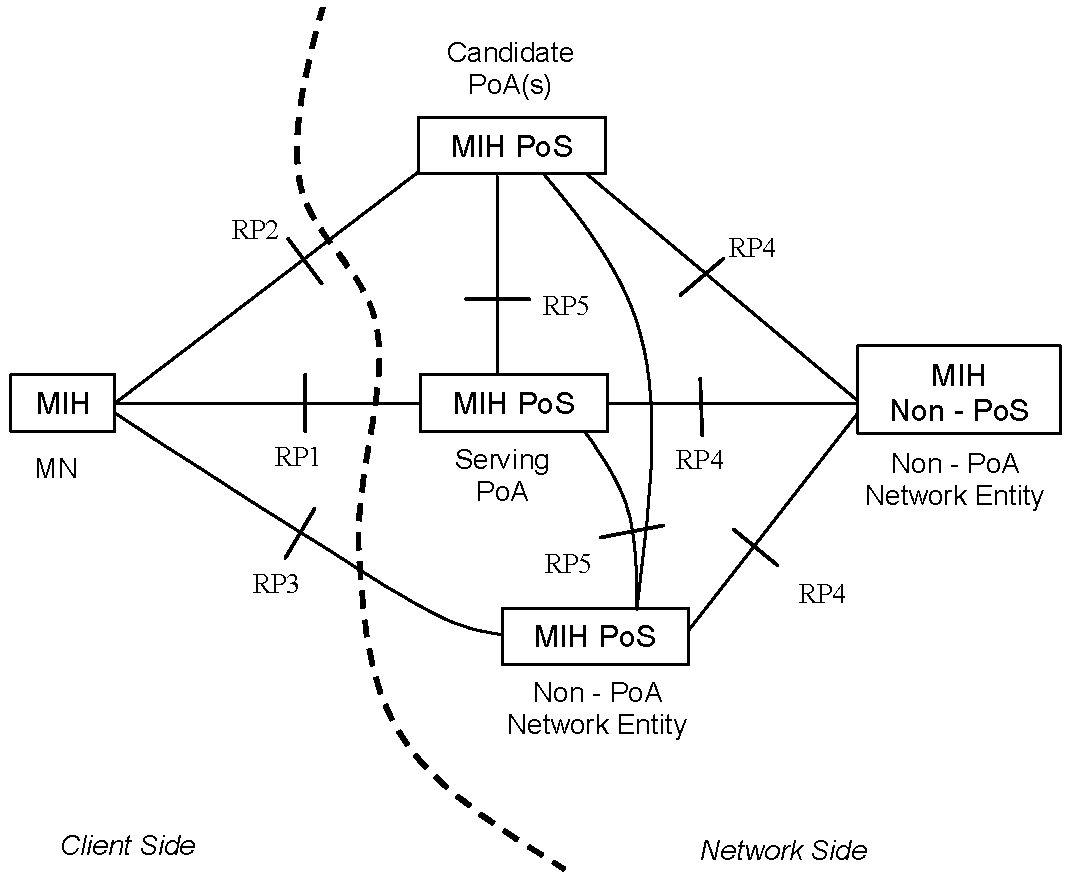
The following subclause describes the key points with regards to communication between different MISF entities in the MN and the network. The reference points in this subclause (5.4) are for illustration only. This subclause does not define any specific deployed network system architecture.

MISF communication model

MIS Functions communicate with each other for various purposes. The MN exchanges MIS information with its MIS point of service (PoS). The MISF in any Network Entity becomes an MIS PoS when it communicates directly with an MN-based MISF. When an MISF in a Network Entity does not have a direct connection to the MN, it does not act as an MIS PoS for that particular MN. However the same MIS Network Entity can still act as MIS PoS for a different MN.

An MN can have multiple L2 interfaces. However, MISF communication need not take place on all L2 interfaces of an MIS-capable MN. As an example, on an MIS-capable MN with three L2 interfaces, namely IEEE 802.11, IEEE 802.16, and IEEE 802.3, the IEEE 802.3 interface might be used only for system administration and maintenance operations, while the IEEE 802.11 and IEEE 802.16 interfaces might engage in the provisioning of MISF services. The MN can use L2 transport for exchanging MIS information with an MIS PoS that resides in the same Network Entity as its Network PoA. The MN can use L3 transport for exchanging MIS information with an MIS PoS that does not reside in the same Network Entity as its Network PoA. The framework supports use of either L2 or L3 mechanisms for communication among MIS network entities.

Figure 2 shows the MISF communication model. The model shows MISFs in different roles and the communication relationships among them. The communication relationship shown in Figure 2 applies only to MISFs. It is important to note that each of the communication relationships in the communication model does not imply a particular transport mechanism. Rather, a communication relationship only intends to show that passing MISF-related information is possible between the two different MISFs. Moreover, each communication relationship shown in the diagram encompasses different types of interfaces, different transport mechanisms used (e.g., L2, L3), and different MISF service related content being passed (e.g., MIIS, MICS, or MIES).



1. —MISF communication model

The communication model assigns different roles to the MISF depending on its position in the system.

1. MISF on the MN
2. MIS PoS on the Network Entity that includes the serving PoA of the MN
3. MIS PoS on the Network Entity that includes a candidate PoA for the MN
4. MIS PoS on a Network Entity that does not include a PoA for the MN
5. MIS non-PoS on a Network Entity that does not include a PoA for the MN

The communication model also identifies the following reference points between different instances of MISFs (see Table 1).

* **Reference point RP1:** Reference point RP1 refers to MISF procedures between the MISF on the MN and the MIS PoS on the Network Entity of its serving PoA. RP1 encompasses communication interfaces over both L2 and L3 and above. MISF content passed over RP1 are related to MIIS, MIES, or MICS.
* **Reference point RP2:**Reference point RP2 refers to MISF procedures between the MISF on the MN and the MIS PoS on the Network Entity of a candidate PoA. RP2 encompasses communication interfaces over both L2 and L3 and above. MISF content passed over RP2 are related to MIIS, MIES, or MICS.
* **Reference point RP3:** Reference point RP3 refers to MISF procedures between the MISF on the MN and the MIS PoS on a non-PoA Network Entity. RP3 encompasses communication interfaces over L3 and above and possibly L2 transport protocols like Ethernet bridging, or multi-protocol label switching (MPLS). MISF content passed over RP3 are related to MIIS, MIES, or MICS.
* **Reference point RP4:** Reference point RP4 refers to MISF procedures between an MIS PoS in a Network Entity and an MIS non-PoS instance in another Network Entity. RP4 encompasses communication interfaces over L3 and above. MISF content passed over RP4 are related to MIIS, MIES, or MICS.
* **Reference point RP5:** Reference point RP5 refers to MISF procedures between two MIS PoS instances in different Network Entities. RP5 encompasses communication interfaces over L3 and above. MISF content passed over RP5 are related to MIIS, MIES, or MICS.

1. —Summary of reference points

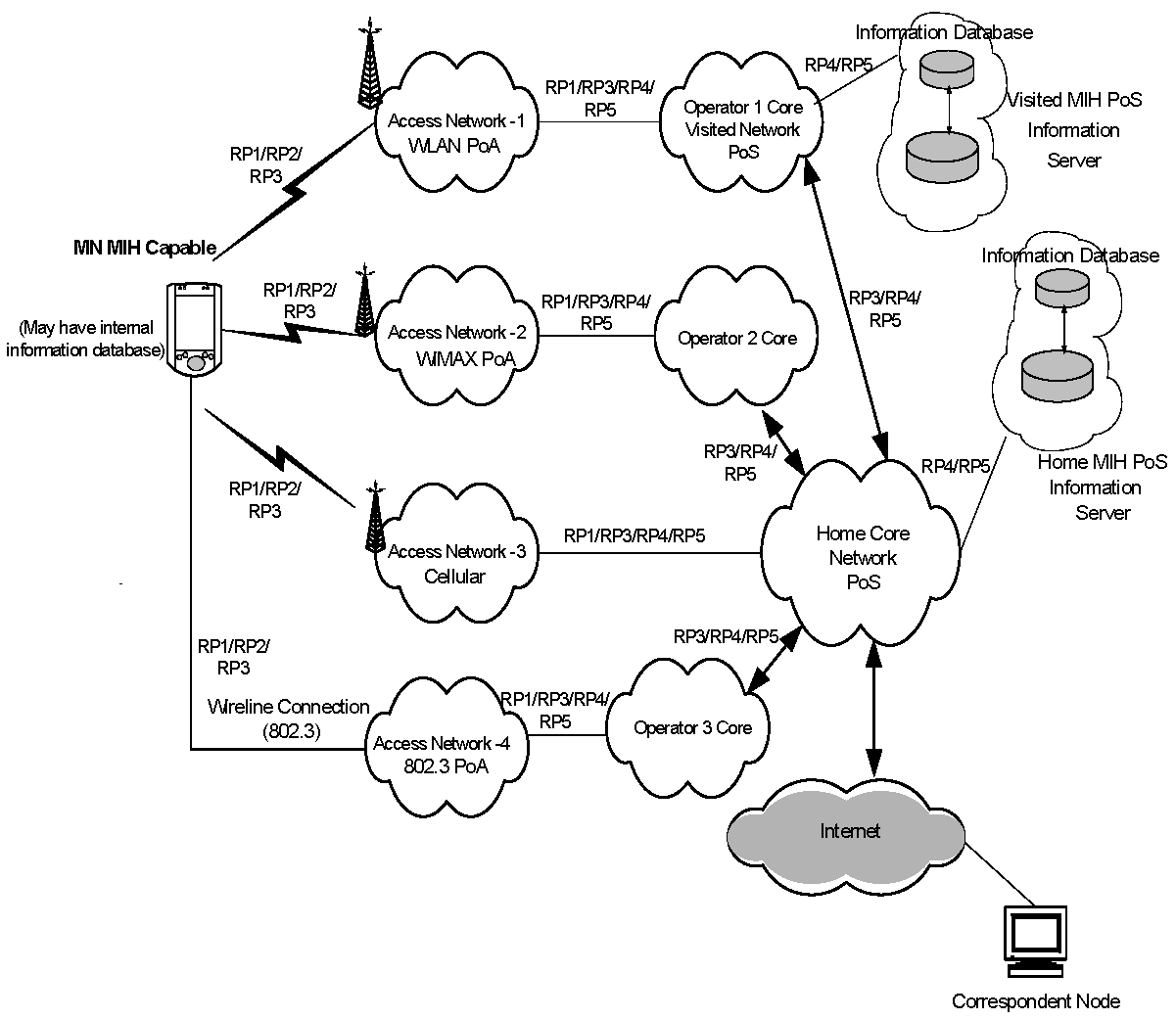
|  |  |
| --- | --- |
| **Reference point** | **Description** |
| RP1 | Between the MISF on an MN and an MIS PoS on the Network Entity of the serving PoA. |
| RP2 | Between the MISF on an MN and an MIS PoS on the Network Entity of the candidate PoA. |
| RP3 | Between the MISF on an MN and an MIS PoS on a non-PoA network entity. |
| RP4 | Between an MISF PoS and an MIS non-PoS instance in different Network Entities. |
| RP5 | Between two MIS PoS instances in different Network Entities. |

All reference point definitions are within the scope of this standard. Annex D provides a mapping of various MIS messages to the reference points.

A deployment example for the MIS services

A network model including MIS services is shown in Figure 3 to better illustrate the MIS Reference Points. Moving from left to right, the model includes an MIS-capable mobile node (MN, far left) that supports multiple wired and wireless access technologies. The model assumes that the serving network either operates multiple link-layer technologies or allows its user to roam into other networks when a service level agreement (SLA) in support of inter-working has been established.

The model illustrates access networks that are connected in some loose, serial way to a given core network (i.e., Core Operator 1, 2, or 3). Also depicted is an access network that is more tightly coupled (Access Network-3). Not depicted in Figure 3, an access network can also connect to a core network via the Internet. Each Core Operator network (1, 2, or 3) might represent a service provider, corporate intranet provider, or just another part of the visited or home access. In this depicted model, the provisioning provider is operating Access Network-3, which couples the terminal to the core (labeled Home Core Network) via RP1. At any given point in time, the subscriber’s serving network can be the home subscriber network or a visited network.



1. —Example of network model with MIS services

The network providers offer MIS services in their access networks (Access Network 1 to 4) in order to facilitate heterogeneous handovers into their networks. Each access technology either advertises its MIS capability or responds to MIS service discovery. Each service provider for these access networks allows access to one or more MIS Points of Service (PoS) node(s). These PoS nodes provide some or all of the MIS services as determined during the MIS capabilities discovery. The PoS location varies based on the operator deployment scenario and the technology-specific MIS architecture.

An MIS PoS resides next to, or is co-located with, the point of attachment (PoA) node in the access network (e.g., Access Network 1, 2, 4). Alternatively the PoS can reside deeper inside the access or core networks (e.g., Access Network 3). As shown in Figure 3, the MIS entity in the MN can communicate with MIS network entities using reference points RP1, RP2, or RP3 over any of the available access networks. If the PoA in the serving access network has a co-located MISF, the RP1 reference point terminates at the PoA that is also the PoS (MN to Access Network 1, 2, 4 of the model can all be RP1). In that case, an RP3 reference point would be terminated at any non-PoA (illustrated by MN connectivity to Access Networks 1, 2, 4). MIS events originate at both sides of an active RP1 link. The MN is typically the first node to react to these events.

The interaction of visited and home subscriber networks could be either for control and management  
purposes or for data transport purposes. It is also possible that due to roaming or SLA agreements, the home subscriber network allows the MN to access the public Internet directly through a visited network. As illustrated, two MIS network entities communicate with each other via RP4 or RP5 reference points. The MIS-capable PoA communicate with other MIS network entities via RP4 and RP5 reference points. The MIS-capable MN have MIS communication with other PoA in the candidate access networks via the RP2 reference point to obtain Information Services about the candidate network.

With regard to the MIS Information Service, visited providers can offer access to their information server located in an MIS PoS node (upper far right). The operator provides the MIIS to MNs so they can obtain pertinent information including, but not limited to, new roaming lists, costs, provider identification information, provider services, priorities, and any other information that would enable the selection and utilization of these services. As illustrated, it is possible for the MN to be pre-provisioned with MIIS data by its provider. It is also possible for the MN to obtain MIS Information Services from any access network of its service provider or from visited networks that maintain SLA agreements with the MN’s service provider. MIIS can also be available from another overlapping or nearby visited network, using that network’s MIIS point of service. The serving network utilizes RP4 and RP5 interfaces to access other MIS entities. As an example, in Figure 3 the home subscriber network accesses its own MIS information server or core operator 1 (visited network) MIS information server.

* 1. MISF reference models for link-layer technologies

The MISF provides asynchronous and synchronous services through well-defined service access points for MIS users. The following subclauses (5.5.1 through 5.5.7) describe the reference models for various link- layer technologies with MIS functionality.

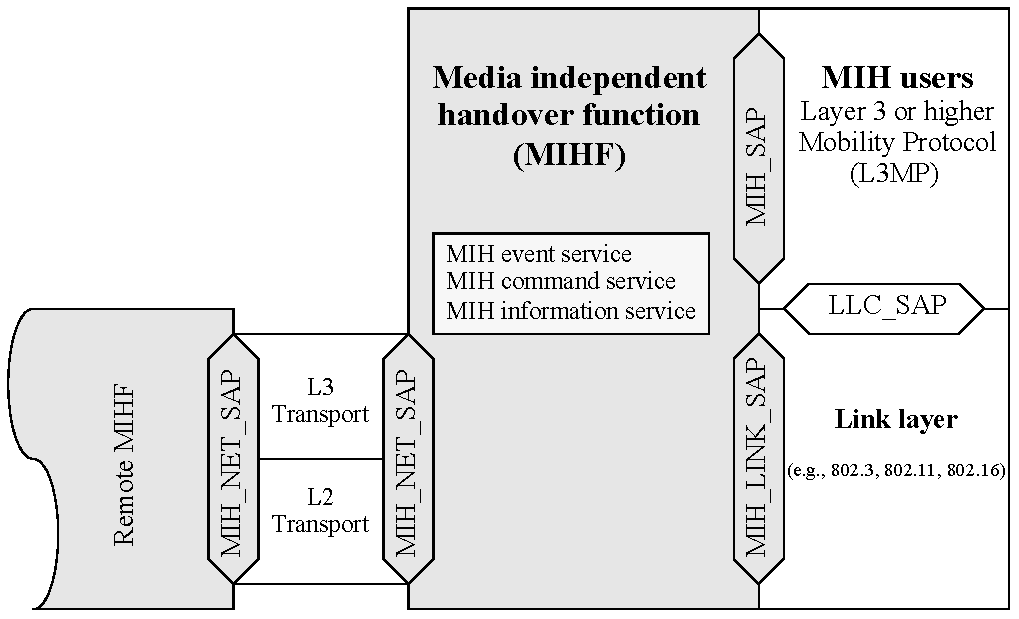
IEEE 802 architectural considerations

The MIS reference models for different IEEE 802 technologies and the general MIS framework is designed to be consistent with the IEEE 802 Architecture for different link-layer technologies. The MIS Function is a management entity that obtains link-layer information from lower layers of different protocol stacks and also from other remote nodes. The MIS Function coordinates handover decision making with other peer MIS Functions in the network.

The MIS Protocol provides the capability for transferring MIS messages between peer MIS Function entities at L2 or at L3. These messages transfer information about different available networks and also provide network switching and handover capability across different networks. The MIS protocol encompasses IEEE 802 technologies such as IEEE 802.11 and IEEE 802.16 and also other non-IEEE 802 technologies such as those specified by 3GPP and 3GPP2 standards. In this sense, the MIS Protocol has different scope and functionality than the Link Layer Discovery Protocol (LLDP) as specified by IEEE Std 802.1ABTM [B18].

General MISF reference model and SAPs

Figure 4 illustrates the position of the MISF in a protocol stack and the interaction of the MISF with other elements of the system. All exchanges between the MISF and other functional entities occur through service primitives, grouped in service access points (SAPs).



1. —General MISF reference model and SAPs

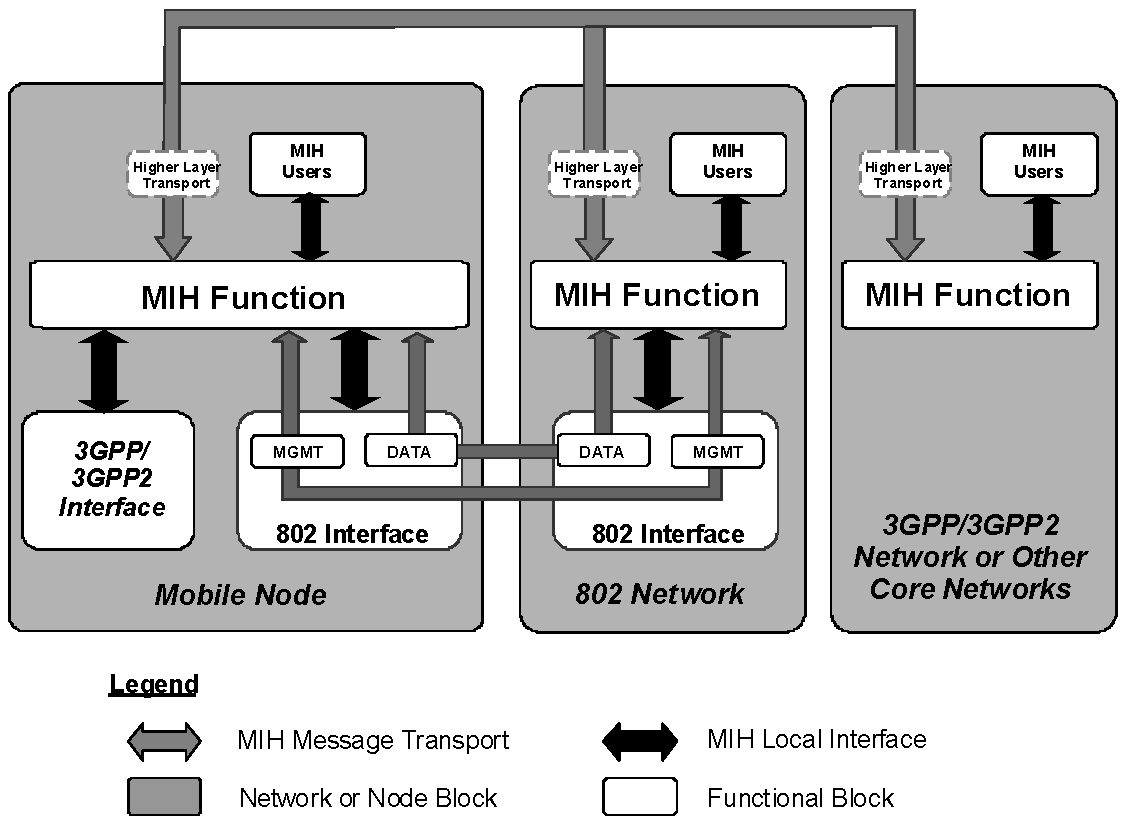
The media agnostic general MIS reference model includes the following SAPs:

1. MIS\_SAP: Media independent interface of MISF with the upper layers of the protocol stack.
2. MIS\_LINK\_SAP: Abstract media dependent interface of MISF with the lower layers of the media- specific protocol stacks.
3. MIS\_NET\_SAP: Abstract media dependent interface of MISF that provides transport services over the data plane on the local node, supporting the exchange of MIS information and messages with the remote MISF. For all transport services over L2, the MIS\_NET\_SAP uses the primitives specified by the MIS\_LINK\_SAP.

In the media-specific reference models, the media independent SAP (MIS\_SAP) always maintains the same name and same set of primitives. The media dependent SAP (which is a technology-specific instantiation of the MIS\_LINK\_SAP), assumes media-specific names and sets of primitives, often reusing names and primitives that already exist in the respective media-specific existing lower-layer SAPs. Primitives defined in MIS\_LINK\_SAP result in amendments to media-specific SAPs due to additional functionality being defined for interfacing with the MISF. All communications of the MISF with the lower layers of media- specific protocol stacks take place through media-specific instantiations of MIS\_LINK\_SAP.

The message exchanges between peer MISF instances, in particular the type of transport that they use, are sensitive to several factors, such as the nature of the network nodes that contain the peer MISF instances (whether or not one of the two is an MN or a PoA), the nature of the access network (whether IEEE 802 or 3G cellular), and the availability of MIS capabilities at the PoA.

Figure 5 presents a summary of the types of relationships that can exist between the MISF and other functional components in the same network node.



1. —Types of MISF relationship

The general MIS reference model in Figure 4 enables a simple representation of the broad variety of MISF relationships shown in Figure 5. In the model, a mobility-management protocol stack is logically identified within each network node that includes an MISF instance. The provided abstraction makes it easy to isolate and represent the MIS relationships with all pre-existing functional entities within the same network node. Such relationships are both internal (with functional entities that, just like the MISF, share the logical inclusion in the mobility-management protocol) and external (with functional entities that belong to other planes).

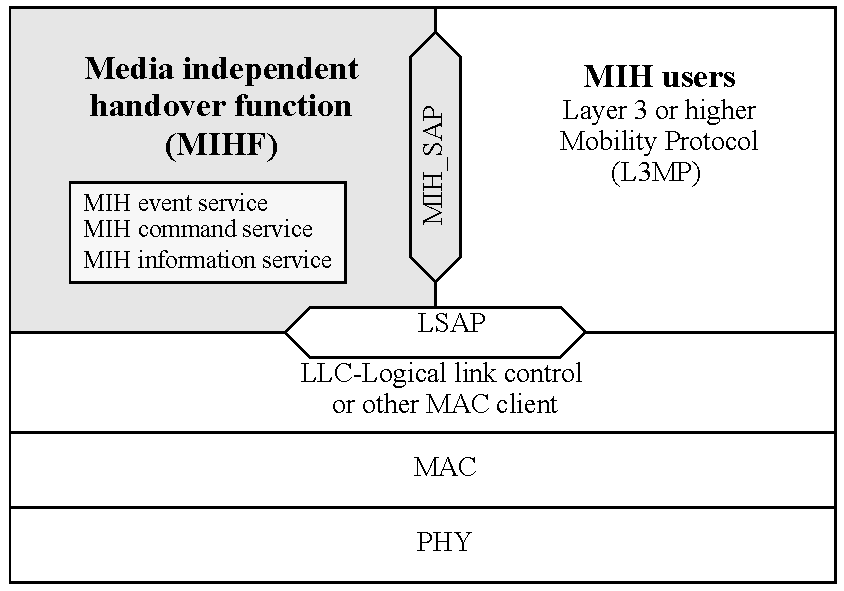
Figure 5 shows how an MIS-enabled MN communicates with an MIS-enabled network. The gray arrows show the MIS signaling over the network, whereas the black arrows show local interactions between the MISF and lower and higher layers in the same network or node block. For a more detailed view of local interactions, please refer to technology-specific reference models and service access point in 5.5.3 through 5.5.7.

When connected to an IEEE 802 network, an MN directly uses L2 for exchanging MIS signaling, as the peer MISF can be embedded in a PoA. The MN does this for certain IEEE 802 networks even before being authenticated with the network. However, the MN can also use L3 for exchanging MIS signaling, for example in cases where the peer MISF is not located in the PoA, but deeper in the network.

When connected to a 3GPP or 3GPP2 network, an MN uses L3 transport to conduct MIS signaling.

MISF reference model for IEEE 802.3

The MISF reference model for IEEE 802.3 is illustrated in Figure 6. The transport of MISF services is supported over the data plane by use of existing primitives defined by the logical link control service access point (LSAP). There are no amendments specified in IEEE Std 802.3 to support any link services defined over the MIS\_LINK\_SAP in this specification.



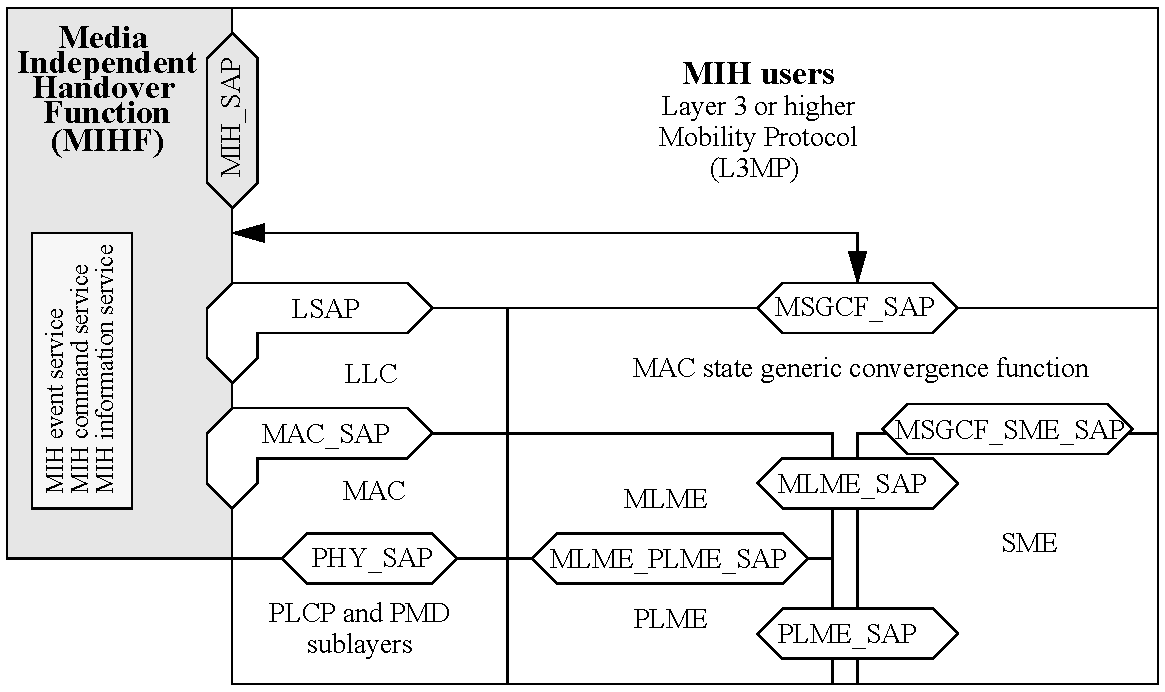
1. —MIS reference model for IEEE 802.3

MISF reference model for IEEE 802.11

Figure 7 shows the MISF reference model for IEEE 802.11. The payload of MISF services over IEEE 802.11 is carried either in the data frames by using existing primitives defined by the LSAP or by using primitives defined by the MAC State Generic Convergence Function (MSGCF) service access point (SAP) (MSGCF\_SAP). The MSGCF has access to all management primitives and provides services to higher layers.

It should be noted that sending MISF payload over the LSAP is allowed only after successful authentication and association of the station to the access point (AP). Moreover, before the station has authenticated and associated with the AP, only MIS Information Service and MIS capability discovery messages can be transported over the MSGCF\_SAP.

The MIS\_SAP specifies the interface of the MISF with MIS users.



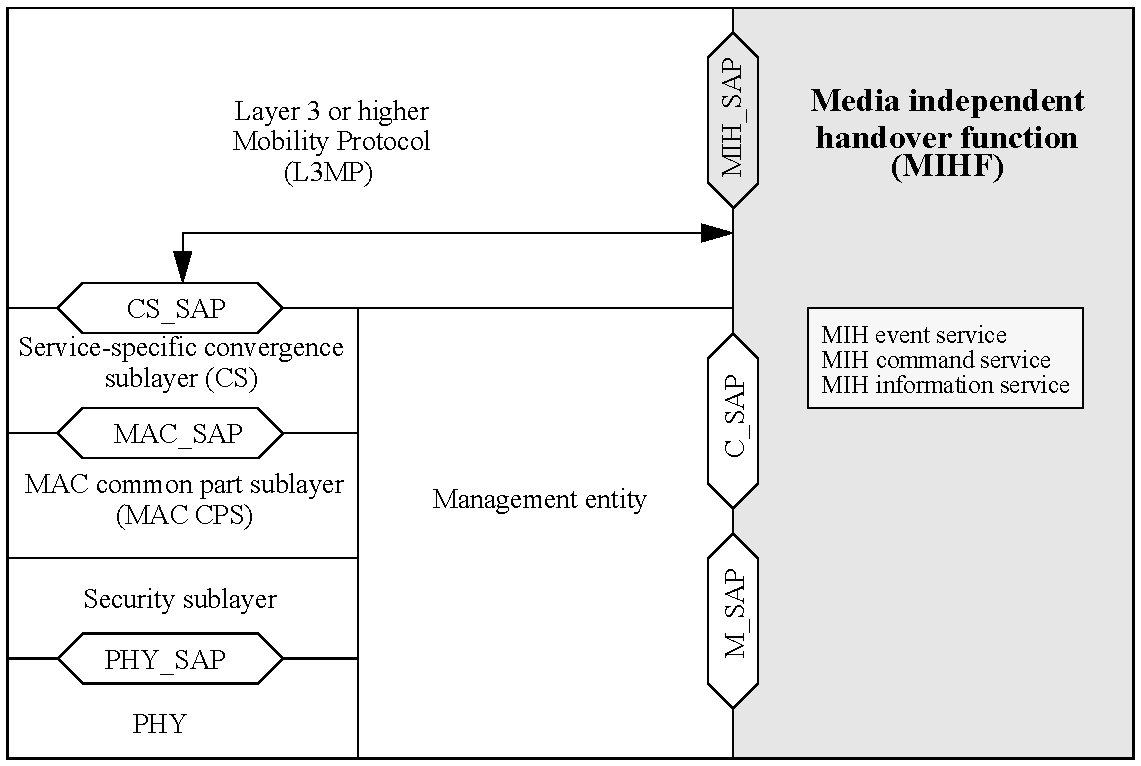
1. —MIS reference model for IEEE 802.11

MISF reference model for IEEE 802.16

Figure 8 shows the MISF for IEEE 802.16 based systems. The Management SAP (M\_SAP) and Control SAP (C\_SAP) are common between the MISF and Network Control and Management System (NCMS).

The M\_SAP specifies the interface between the MISF and the management plane and allows MISF payload to be encapsulated in management messages (such as MOB\_MIS-MSG defined in IEEE P802. 16g [B2 1]). The primitives specified by M\_SAP are used by an MN to transfer packets to a base station (BS), both before and after it has completed the network entry procedures. The C\_SAP specifies the interface between the MISF and control plane. M\_SAP and C\_SAP also transport MIS messages to peer MISF entities. The Convergence Sublayer SAP (CS\_SAP) is used to transfer packets from higher layer protocol entities after appropriate connections have been established with the network.

The MIS\_SAP specifies the interface of the MISF with other higher layer entities such as transport layer, handover policy engine, and layer 3 mobility protocol.

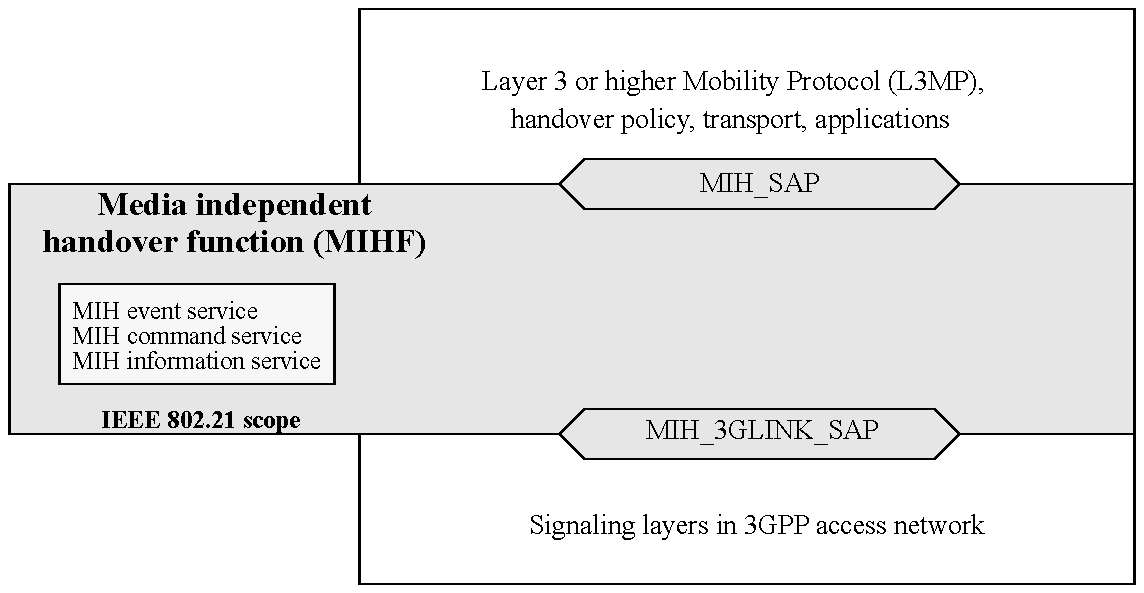


1. —MIS reference model for IEEE 802.16

In this model, C\_SAP and M\_SAP provide link services defined by MIS\_LINK\_SAP, C\_SAP provides services before network entry, while CS\_SAP provides services over the data plane after network entry.

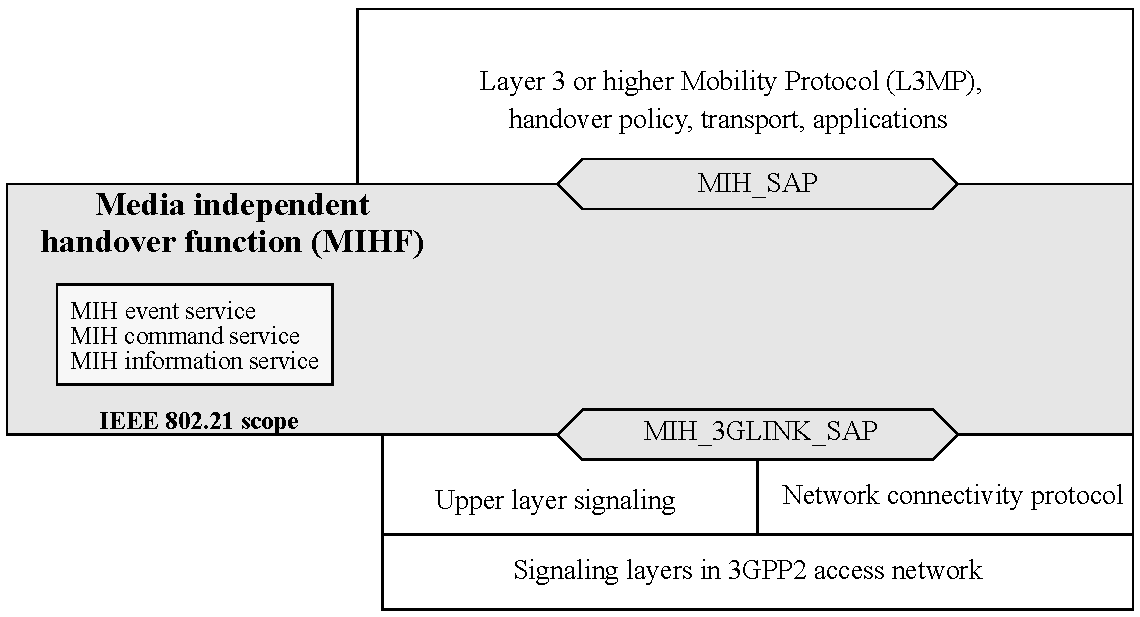
MISF reference model for 3GPP

Figure 9 illustrates the interaction between the MISF and the 3GPP-based systems. The MISF services are specified by the MIS\_3GLINK\_SAP. However, no new primitives or protocols need to be defined in the 3GPP specification for accessing these services. The MISF services are mapped to existing 3GPP signaling functions (see Table E.3 in Annex E). The architectural placement of the MISF is also decided by the 3GPP standard. Figure 9 is for illustrative purposes only and should not constrain implementations.



1. —MIS reference model for 3GPP systems

Figure 10 illustrates the interaction between IEEE 802.21 services and 3GPP2 based systems. IEEE 802.21 services are accessed through the MIS\_3GLINK\_SAP. However note that no new primitives or protocols need to be defined within the 3GPP2 specification. Instead, a mapping between IEEE 802.21 link-layer primitives and 3GPP2 primitives as defined in IETF RFC 1661 and 3GPP2 C.S0004-D is already established. Primitive information available from Upper Layer Signaling and Point-to-Point Protocol (PPP) can be directly used by mapping LAC SAP and PPP SAP primitives to IEEE 802.21 service primitives in order to generate an event.



1. —MIS reference model for 3GPP2 systems

This mapping is illustrated in Table E.3, which provides an example of how 3GPP and 3GPP2 primitives can be mapped to IEEE 802.21 primitives. For example, events received from the Upper Layer Signaling through the LAC layer SAP such as “L2.Condition.Notification” can be mapped and generated through the MIS\_3GLINK\_SAP as a Link\_Up, Link\_Down, or Link\_Going\_Down. Likewise, events generated at the PPP SAP within the PPP layer, such as LCP-Link-Up or IPCP\_LINK\_OPEN, could be mapped and generated through the MIS\_3GLINK\_SAP as a Link\_Up event.

It is noteworthy that there will be no direct communication between the 3GPP2 PHY and MAC layers with the MISF. The architectural placement of any MISF is left to 3GPP2. Figure 10 is for illustrative purposes only and should not constrain implementations.

* 1. Service access points (SAPs)

General

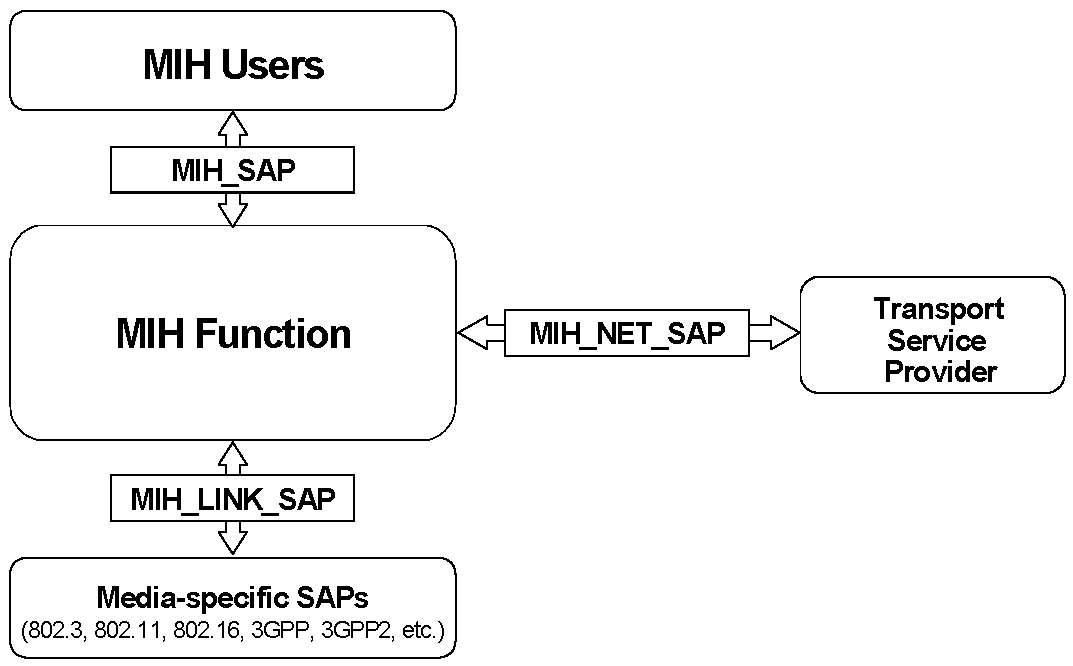
The MISF interfaces with other layers and functional planes using service access points (SAPs). Each SAP consists of a set of service primitives that specify the interactions between the service user and provider.

The specification of the MISF includes the definition of SAPs that are media independent and recommendations to define or extend other SAPs that are media dependent. Media independent SAPs allow the MISF to provide services to the upper layers of the mobility-management protocol stack, the network management plane, and the data bearer plane. The MIS\_SAP and associated primitives provide the interface from MISF to the upper layers of the mobility-management protocol stack. Upper layers need to subscribe with the MISF as users to receive MISF generated events and also for link-layer events that originate at layers below the MISF but are passed on to MIS users through the MISF. MIS users directly send commands to the local MISF using the service primitives of the MIS\_SAP. Communication between two MISFs relies on MIS protocol messages.

Media dependent SAPs allow the MISF to use services from the lower layers of the protocol stack and their management planes. All inputs (including the events) from the lower layers of the protocol stack into the MISF are provided through existing media-specific SAPs such as MAC SAPs, PHY SAPs, and logical link control (LLC) SAPs. Link Commands generated by the MISF to control the PHY and MAC layers during the handover are part of the media-specific MAC/PHY SAPs and are already defined elsewhere.

Figure 11 shows the key MISF-related SAPs for different networks, which are as follows:

1. The MIS\_SAP specifies a media independent interface between the MISF and upper layers of the mobility management protocol stack. The upper layers need to subscribe with the MISF as users to receive MISF-generated events and also for link-layer events that originate at layers below the MISF but are passed on to MISF users through the MISF. MISF users directly send commands to the local MISF using the service primitives of the MIS\_SAP.
2. The MIS\_LINK\_SAP specifies an abstract media dependent interface between the MISF and lower layers media-specific protocol stacks of technologies such as IEEE 802.3, IEEE 802.11, IEEE 802.16, 3GPP, and 3GPP2. For different link-layer technologies, media-specific SAPs provide the functionality of MIS\_LINK\_SAP. Amendments are suggested to the respective media-specific SAPs to provide all the functionality as described by MIS\_LINK\_SAP.
3. The MIS\_NET\_SAP specifies an abstract media dependent interface of the MISF that provides transport services over the data plane on the local node, supporting the exchange of MIS information and messages with remote MISFs.



1. —Relationship between different MISF SAPs

Media dependent SAPs

* + - * 1. General

Each link-layer technology specifies its own technology-dependent SAPs. For each link-layer technology, the MIS\_LINK\_SAP maps to the technology-specific SAPs.

* + - * 1. MIS\_LINK\_SAP

This SAP defines the abstract media dependent interface between MISF and different link-layer technologies. Amendments are suggested for different layer technology-specific SAPs based on the definition of this particular SAP.

* + - * 1. MIS\_NET\_SAP

MIS\_NET\_SAP defines the abstract media dependent interface of the MISF that provides transport services over the data plane on the local node, supporting the exchange of MIS information and messages with remote MISFs. For L2, this SAP uses the primitives provided by MIS\_LINK\_SAP.

* + - * 1. MLME\_SAP

This SAP defines the interface between the MISF and the management plane of an IEEE 802.11 network. This SAP is used for sending MIS messages between the MISF and local link-layer entities, as well as between peer MISF entities.

* + - * 1. C\_SAP

The C\_SAP, defined in IEEE Std 802.16, provides the interface between the MISF and the IEEE 802.16 control plane. This SAP is used for MIS exchanges between the MISF and the lower layers of the management plane (as part of the IEEE 802.16 instantiation of the MIS\_LINK\_SAP).

* + - * 1. M\_SAP

The M\_SAP, defined in IEEE Std 802.16, provides the interface between the MISF and the IEEE 802.16 management plane functions.

* + - * 1. MSGCF\_SAP

This SAP, defined in IEEE P802.1 1u/D3.0, provides services to MISF based on the IEEE 802.11 MAC state machines and interactions between the IEEE 802.11 sublayers.

* + - * 1. MIS\_3GLINK\_SAP

This SAP works as an umbrella that defines the interface between the MISF and the different protocol elements of the cellular systems. The existing service primitives or media-specific SAPs as defined in 3GPP and 3GPP2 specifications are directly mapped to MISF services, and hence no new primitives need to be defined in these specifications. Table E.3 lists this mapping.

* + - * 1. LSAP

The logical link control service access point (LSAP), defined in IEEE Std 802.2, provides the interface between the MISF and the Logical Link Control sublayer in IEEE 802.3 and IEEE 802.11 networks. This SAP is used for local MIS exchanges between the MISF and the lower layers and for the L2 transport of MIS messages across IEEE 802 access links.

* + - * 1. CS\_SAP

The CS\_SAP, defined in IEEE Std 802.16, provides the interface between the MISF and the service-specific Convergence Sublayer in IEEE 802.16 networks. This SAP is used for the L2 transport of MIS messages across IEEE 802.16 access links.

Media independent SAP: MIS\_SAP

The MIS\_SAP defines the media independent interface between the MISF and MIS users such as an upper  
layer mobility protocol or a handover function that might reside at higher layers or a higher layer transport  
entity as well. The definition of the MIS\_SAP is required to define the scope and functionality of the MISF.

* 1. MIS protocol

General

MIS Protocol defines the format of messages (i.e., MISF packet with header and payload) that are exchanged between remote MISF entities and the media independent mechanisms that support the delivery of these messages.

Ethertype use and encoding

All MIS protocol data units (PDUs) shall be identified using the MIS protocol Ethertype specified in Table 2.

1. —MIS protocol Ethernet type

|  |  |
| --- | --- |
| Assignment | Valuea |
| MIS protocol Ethernet type | 89–17 |

aThis Ethertype value is expressed using the hexadecimal represen­tation defined in IEEE Std 802.

Transport considerations

MIS Protocol messages are sent over the data plane by use of a suitable transport mechanism at both layer 2 and layer 3. Layer 3 transport is supported using transmission control protocol (TCP) / user datagram protocol (UDP) / stream control transmission protocol (SCTP) protocols over IP. Layer 2 transport is supported with the EtherType value set to that for MIS Protocol. The data plane is available for transport after the MN has authenticated with the access network. In case of IEEE 802.11 and IEEE 802.16 networks, MIS Protocol messages can also be sent before authentication over the management plane by using respective media specific MAC management frames.

The generic MAC service with IEEE 802.1X

The generic MAC service in both IEEE 802.3 network and IEEE 802.11 robust security network association (RSNA) networks (which use IEEE Std 802. 1X-2004 port-based network access control), goes through the controlled port after authentication and association. The uncontrolled port is in open access mode to allow only exchange of messages to perform authentication and secure connection association, whereas the controlled port is blocked until authentication and association are successful.

The MIS messages that pass through the LSAP are distinguished from other protocols with an EtherType value of MIS protocol ethernet type in the LLC header.

When the MIS protocol is used across IEEE 802.11 networks, MIS frames may be exchanged when the STA has successfully authenticated and associated to the IEEE 802.11 access point.

* + - * 1. Controlled port unblocked state: LSAP transport

After successful authentication and association, the controlled port is unblocked to the transport of authenticated messages. The MIS messages are then encapsulated into LLC protocol with EtherType value of MIS protocol to pass through the controlled port. When LSAP receives MAC frames from the LLC layer, it checks the EtherType of each frame to determine whether to send the frame to MISF protocol or to other protocols.

* + - * 1. Controlled port blocked state

Until authentication has been completed, the controlled port is in blocked state so that MIS messages cannot go through. However, in IEEE 802.11 and IEEE 802.16 networks, MIS messages (MIS\_messages are limited to Information Service Query request/response, Event Service, and Command Service capability discovery only) can be transported via the management plane prior to authentication.

1. MISF services
   1. General

The MISF provides the media independent event service, the media independent command service, and the media independent information service that facilitate handovers across heterogeneous networks. Clause 6 provides a general description of these services. These services are managed and configured through service management primitives, as discussed in 6.2.

* 1. Service management

General

Prior to providing the MIS services from one MISF to another, the MIS entities need to be configured properly. This is done through the following service management functions:

* MIS capability discovery
* MIS registration
* MIS service access authentication
* MIS event subscription

In order to know the services that are supported by an MIS peer, the MIS node performs MIS capability discovery. The MIS node performs MIS capability discovery with different MIS peers in order to decide which one to register with.

Service management primitives

Table 3 defines the set of service management primitives. A primitive is marked as local only (L), remote only (R), or local and remote (L, R), indicating whether it can be invoked by a local MIS user, a remote MIS user, or both, respectively.

1. —Service management primitives

|  |  |  |  |
| --- | --- | --- | --- |
| **Service management primitive** | **(L) ocal, (R) emote** | **Defined in** | **Comments** |
| MIS\_Capability\_Discover | L, R | 7.4.1 | Discover the capabilities of a local or remote MISF. |
| MIS\_Register | R | 7.4.2 | Register with a remote MISF. |
| MIS\_DeRegister | R | 7.4.3 | Deregister from a remote MISF. |
| MIS\_Event\_Subscribe | L, R | 7.4.4 | Subscribe for one or more MIS events with a local or remote MISF. |
| MIS\_Event\_Unsubscribe | L, R | 7.4.5 | Unsubscribe for one or more MIS events from a local or remote MISF. |
| MIS\_Push\_Key | L, R | 7.4.27 | Install a key in a remote PoA |
| MIS\_LL\_Auth | L, R | 7.4.28 | Carry out a proactive auth entication over MIS between the MN and the PoS using link layer frames. |

MIS capability discovery

The MIS capability discovery procedure is used by an MIS user to discover a local or remote MISF’s  
capabilities in terms of MIS services (Event Service, Command Service, and Information Service). MIS capability discovery is performed either through the MIS protocol or through media-specific mechanisms (i.e., IEEE 802.11 Beacon frames, IEEE 802.16 downlink channel descriptor (DCD), IEEE 802.11 management frames, or IEEE 802.16 management messages).

MIS registration

MIS registration is defined as a means of requesting access to specific MIS services. For example, in a network controlled inter-technology handover framework, MIS registration can be used by an MN to declare its presence to a selected MIS PoS. MIS Registration is mandatory for use with the MIS Command Service and the push mode of the MIS Information Service.

MIS event subscription

The MIS event subscription mechanism allows an MIS user to subscribe for a particular set of events that originates from a local or remote MISF. See 6.3.2 for a more detailed description of MIS event subscription.

Network communication

The network communication functions provide transport services over the data plane on the local node, supporting the exchange of MIS information and messages between the local and remote MISF. For transport services over L2, MIS\_NET\_SAP utilizes the primitives specified by the MIS\_LINK\_SAP. For transport services over L3, the primitives are specified by MIS\_NET\_SAP. Please refer to 7.5 for more details on MIS\_NET\_SAP.

* 1. Media independent event service

Introduction

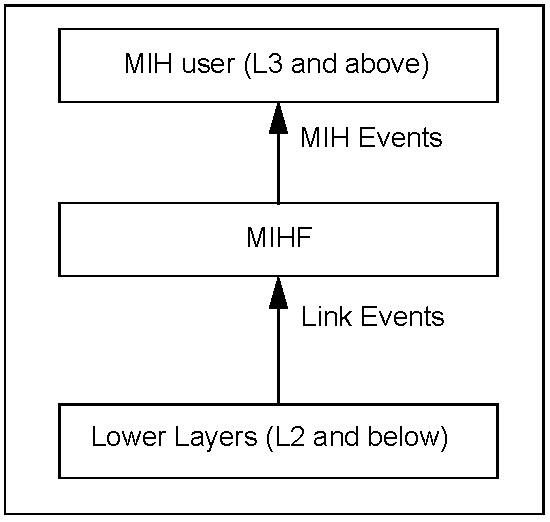
In general, handovers can be initiated either by the MN or by the network. Events relevant to handover originate from MAC, PHY, or MISF at the MN, at the network PoA, or at the PoS. Thus, the source of these events is either local or remote entity. A transport protocol is needed for supporting remote events. Security is another important consideration in such transport protocols.

Multiple higher layer entities can be interested in these events at the same time. Thus, these events can have multiple destinations. Higher layer entities can subscribe to receive event notifications from a particular event source. The MISF can help in dispatching these events to multiple destinations.

These events are treated as discrete events. As such there is no general event state machine. Event notifications are generated asynchronously. Thus, all MIS users and MISFs that want to receive event notifications need to subscribe to particular events.

From the recipient’s perspective, these events are mostly “advisory” in nature and not “mandatory.” The recipient is not obligated to act on these events. Layer 3 and above entities need to deal with reliability and robustness issues associated with these events. Higher layer protocols and other entities can take a more cautious approach when events originate remotely as opposed to when they originate locally. These events can also be used for horizontal handovers.

The Event Service is broadly divided into two categories, Link Events and MIS Events. Both Link and MIS Events traverse from a lower to a higher layer. Link Events are defined as events that originate from event source entities below the MISF and terminate at the MISF. Entities generating Link Events include, but are not limited to, various IEEE 802-defined, 3GPP-defined, and 3GPP2-defined interfaces. Within the MISF, Link Events propagate further, with or without additional processing, to MIS users that have subscribed for the specific events. MIS events are defined as events that originate from within the MISF, or they are Link Events that are propagated by the MISF to the MIS users. This relationship is shown in Figure 12.

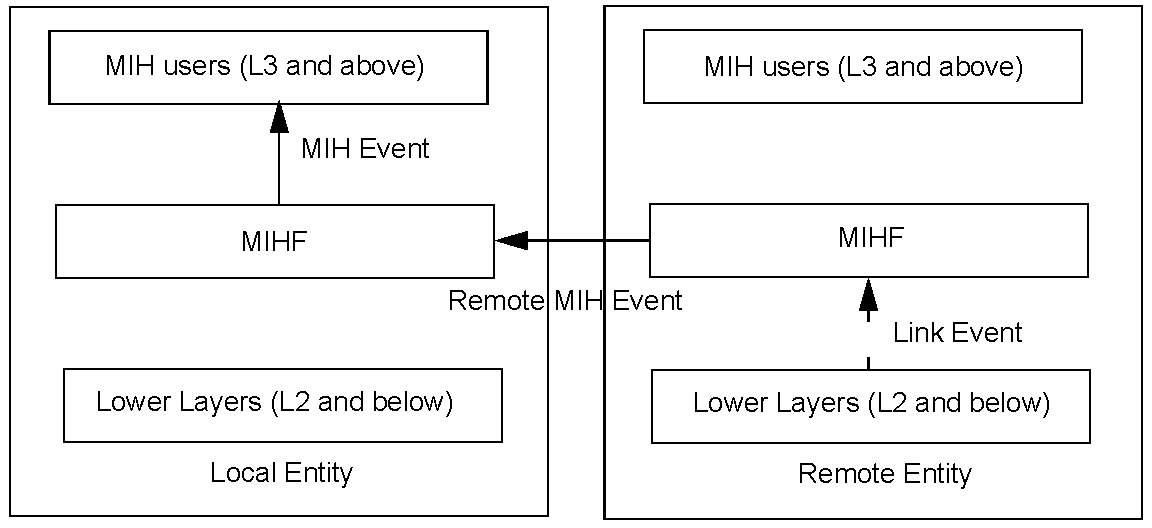


1. —Link events and MIS events

An event can be local or remote; a local event is one that propagates across different layers within the local protocol stack of an MIS entity, while a remote event is one that traverses across the network medium from one MIS entity to another MIS entity.

All Link Events are local in nature and propagate from the local lower layer to the local MISF. MIS Events are local or remote. A remote MIS Event traverses the medium from a remote MISF to the local MISF and is then dispatched to local MIS users that have subscribed to this remote event, as shown in Figure 13.

A Link Event that is received by the MISF can also be sent to a remote MIS entity as a remote MIS Event.



1. —Remote MIS events

Event subscription

* + - * 1. General

Event subscription provides a mechanism for upper layer entities to selectively receive events. Event subscription can be divided into link events subscription and MIS events subscription. Link events subscription is performed by the MISF with the event source entities in order to determine the events that each event source (link) is able to provide. MIS events subscription is performed by upper layer entities with the MISF to select the events to receive. It is possible for upper layer entities to subscribe for all existing events or notifications that are provided by the event source entity even if no additional processing of the event is done by the MISF.

* + - * 1. Link events subscription

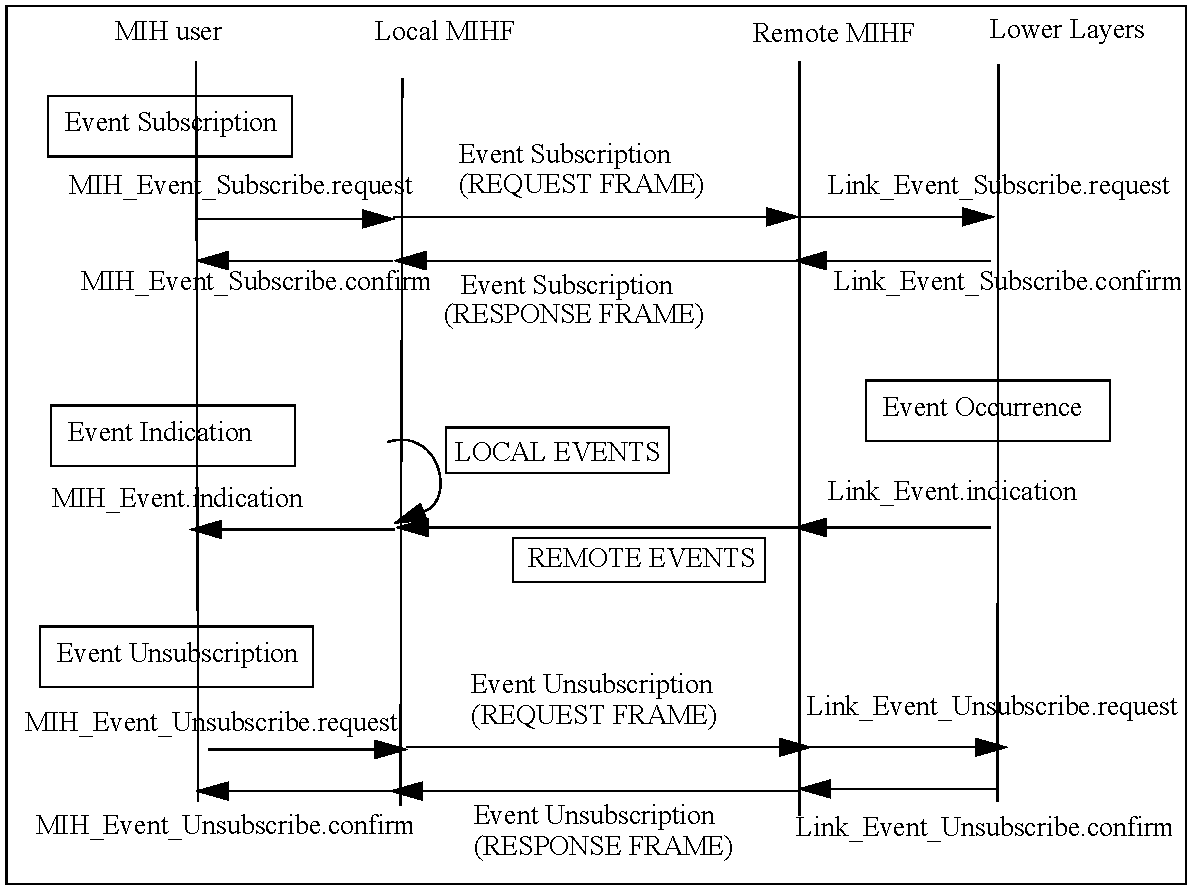
During initialization the MISF actively searches for pre-existing interfaces, devices, and modules that serve as link event sources in the Event service. In addition to the link event source entities that are present during the bootstrapping stage, allowances are made for devices such as hot-plugged interfaces or an external module. The exact description and implementation of such mechanisms is out of the scope of the standard. The MISF subscribes individually with each of these link layers based on user preferences.

* + - * 1. MIS events subscription

MIS users specify a list of events for which they wish to receive notifications from the MISF. For an MIS event that can originate both locally and remotely, an MIS user specifies whether it is subscribing for the local event only, remote event only, or both (which would require two separate subscriptions). If the MIS event that an MIS user wants to subscribe to is not supported or is not available, then the MISF rejects the subscription request and notifies the MIS user accordingly.

Event service flow model

Figure 14 shows the event flow model for link events and MIS events.



1. —MIS events subscription and flow

Link events

The media independent event service supports the following several categories of link events:

1. **MAC and PHY State Change events:** These events correspond to changes in MAC and PHY state. For example, Link\_Up event is a state change event.
2. **Link Parameter events:** These events are due to changes in link-layer parameters. For example, the primitive Link\_Parameters\_Report is a Link Parameter event.
3. **Predictive events:** Predictive events convey the likelihood of a change in the link conditions in the near future based on past and present conditions. For example, decay in signal strength of a wireless local area network (WLAN) indicates a loss of link connectivity in the near future.
4. **Link Handover events:** These events inform upper layers about the occurrence of L2 handovers/ link switches if supported by the given media type.[[12]](#footnote-12)
5. **Link Transmission events:** These events indicate the link-layer transmission status (e.g., success or failure) of upper layer PDUs. This information is used by upper layers to improve buffer management for minimizing the upper layer data loss due to a handover.

For example, the occurrence of a handover of an MN from one access network to another will result in the tear-down of the old link-layer connection between the MN and the source access network and the establishment of a new link-layer connection between the MN and the target access network. When this occurs, some upper layer PDUs still remain buffered at the old link—including PDUs that had been queued at the old link but never been transmitted before the link was torn-down (i.e., unsent PDUs), and PDUs that have been transmitted over the old link but never been fully acknowledged by the upper layer receiver before the link was torn-down (i.e., unacked PDUs). These buffered PDUs will be discarded when the old link is torn-down. As a result, unless the upper layer sender attempts to retransmit them over the new link connection, these upper layer PDUs will never reach the receiver.

Table 4 defines link events.

1. —Link events

|  |  |  |  |
| --- | --- | --- | --- |
| **Link event name** | **Link event type** | **Description** | **Defined in** |
| Link\_Detected | State change | Link of a new access network has been detected. This event is typically generated on the MN when the first PoA of an access network is detected. This event is not generated when subsequent PoAs of the same access network are discovered. | 7.3.1 |
| Link\_Up | State change | L2 connection is established and link is available for use. This event is a discrete event. | 7.3.2 |
| Link\_Down | State change | L2 connection is broken and link is not available for use. This event is a discrete event. | 7.3.3 |
| Link\_Parameters\_Report | Link parameters | Link parameters have crossed pre-specified thresholds. | 7.3.4 |
| Link\_Going\_Down | Predictive | Link conditions are degrading and connection loss is imminent. | 7.3.5 |
| Link\_Handover\_Imminent | Link handover | L2 handover is imminent based on changes in link conditions. | 7.3.6 |
| Link\_Handover\_Complete | Link handover | L2 link handover to a new PoA has been completed. | 7.3.7 |
| Link\_PDU\_Transmit\_Status | Link transmission | Indicate transmission status of a PDU. | 7.3.8 |

In general when a link event occurs due to a change in link condition it is not known at that instant if this would lead to intra-technology handover or inter-technology handover. That determination is done higher up in the protocol stack by the network selection entity based on variety of other factors. As such certain link- layer events such as Link\_Going\_Down leads to either intra-technology or inter-technology handovers. The network selection entity tries to maintain the current connection, by first trying intra-technology handovers and only later on resort to inter-technology handovers.

MIS events

Table 5 defines MIS events. An MIS event is marked as local only (L), remote only (R), or local and remote (L, R), indicating whether it can be subscribed by a local MIS user, a remote MIS user, or both, respectively.

1. —MIS events

|  |  |  |  |
| --- | --- | --- | --- |
| **MIS event name** | **(L) ocal (R) emote** | **Description** | **Defined in** |
| MIS\_Link\_Detected | L, R | Link of a new access network has been detected. This event is typically generated on the MN when the first PoA of an access network is detected. This event is not generated when subsequent PoAs of the same access network are discovered. | 7.4.6 |
| MIS\_Link\_Up | L, R | L2 connection is established and link is available for use. | 7.4.7 |
| MIS\_Link\_Down | L, R | L2 connection is broken and link is not available for use. | 7.4.8 |
| MIS\_Link\_Parameters\_Report | L, R | Link parameters have crossed a specified thresh- old and need to be reported. | 7.4.9 |
| MIS\_Link\_Going\_Down | L, R | Link conditions are degrading and connection loss is imminent. | 7.4.10 |
| MIS\_Link\_Handover\_Imminent | L, R | L2 handover is imminent based on either the changes in the link conditions or additional infor­mation available in the network. For example, the network decides that an application requires a specific QoS that can be best provided by a cer­tain access technology. | 7.4.11 |
| MIS\_Link\_Handover\_Complete | L, R | L2 link handover to a new PoA has been completed. | 7.4.12 |
| MIS\_Link\_PDU\_Transmit\_Status | L | Indicate transmission status of a PDU. | 7.4.13 |

Interaction between MIS events and access routers

Access Router (AR) is a layer 3 (L3) IP router residing in an access network and is connected to one or more PoAs. An AR is the first hop router for an MN.

During heterogeneous handovers an MN can switch from one link technology to another. This will result in a change in the PoA that the MN is connected to. The target PoA and the source PoA may or may not be on the same subnet. In cases where there is a change in subnet, IP packet delivery can be optimized if context (e.g., change in routing information) from the old AR to the target AR is transferred. In such cases, the target router can update its L2 address to IP address mapping.

Link-layer triggers such as Link Going Down and Link Up can be used to indicate departure and arrival of  
MNs at AR(s) and such indications can replace L3 protocol signaling for the same and thus expedite the  
handover process. Layer 3 Mobility management protocols, such as MIP can also benefit from triggers such as Link Going Down. Timely receipt of such triggers by the AR in case of network-controlled handovers can enable MIP signaling to establish the new route to take place in parallel with other handover message exchange and can thus reduce the disruption time in IP packet delivery.

* 1. Media independent command service

Introduction

media independent command service (MICS) refers to the commands sent from MIS users to the lower layers in the reference model. MIS users utilize command services to determine the status of links and/or control the multi-mode device for optimal performance. Command services also enable MIS users to facilitate optimal handover policies. For example, the network initiates and controls handovers to balance the load of two different access networks.

The link status varies with time and MN mobility. Information provided by MICS is dynamic information composed of link parameters such as signal strength and link speed; whereas, information provided by MIIS is less dynamic or static in nature and is composed of parameters such as network operators and higher layer service information. MICS and MIIS information could be used in combination by the MN/network to facilitate the handover.

A number of commands are defined in this standard to allow the MIS users to configure, control, and retrieve information from the lower layers including MAC, Radio Resource Management, and PHY. The commands are classified into two categories: MIS commands and link commands. Figure 15 shows link commands and MIS commands.

The receipt of certain MIS command requests can cause event indications to be generated. The receipt of MIS command requests indicates a future state change in one of the link layers in the local node. These indications notify subscribed MIS users of impending link state changes. This allows MIS users to be better prepared to take appropriate action.

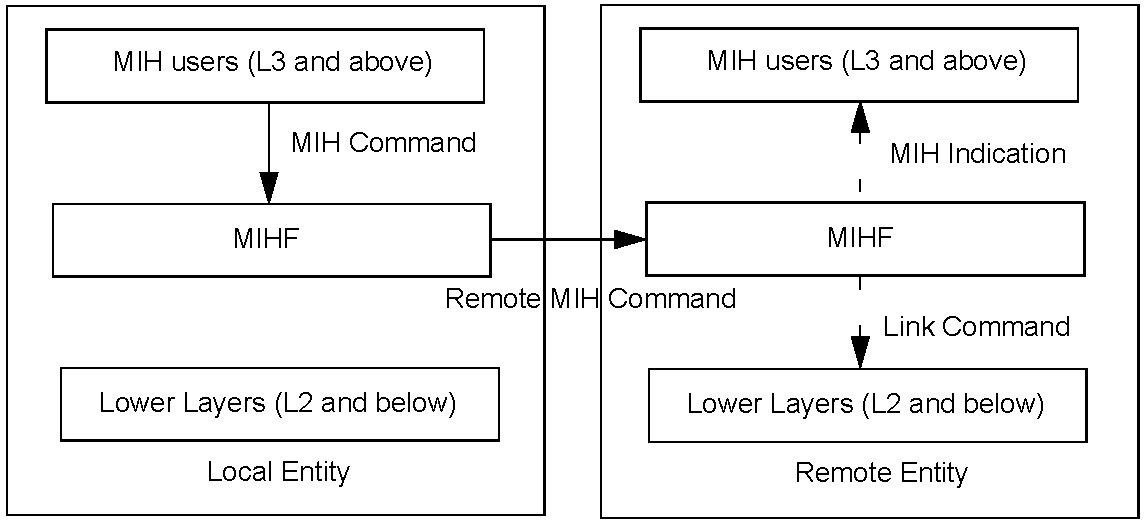
Link commands originate from the MISF and are directed to the lower layers. These commands mainly control the behavior of the lower layer entities. Link commands are local only. Whenever applicable, this standard encourages use of existing media-specific link commands for interaction with specific access networks. New link commands, if required, are defined as recommendations to different link-layer technology standards. It is to be noted that although Link commands originate from the MISF, these commands are executed on behalf of the MIS users.

The MIS commands are generated by the MIS users and sent to the MISF. MIS commands can be local or remote. Local MIS commands are sent by MIS users to the MISF in the local protocol stack.



1. —Link commands and MIS commands

Remote MIS commands are sent by MIS users to the MISF in a peer protocol stack. A remote MIS command delivered to a peer MISF is executed by the lower layers under the peer MISF as a link command; or is executed by the peer MISF itself as an MIS command (as if the MIS command came from an MIS user of the peer MISF); or is executed by an MIS user of the peer MISF in response to the corresponding indication. Often, an MIS indication to a remote MIS user results from the execution of the MIS command by the peer MISF. Figure 16 shows remote MIS commands.

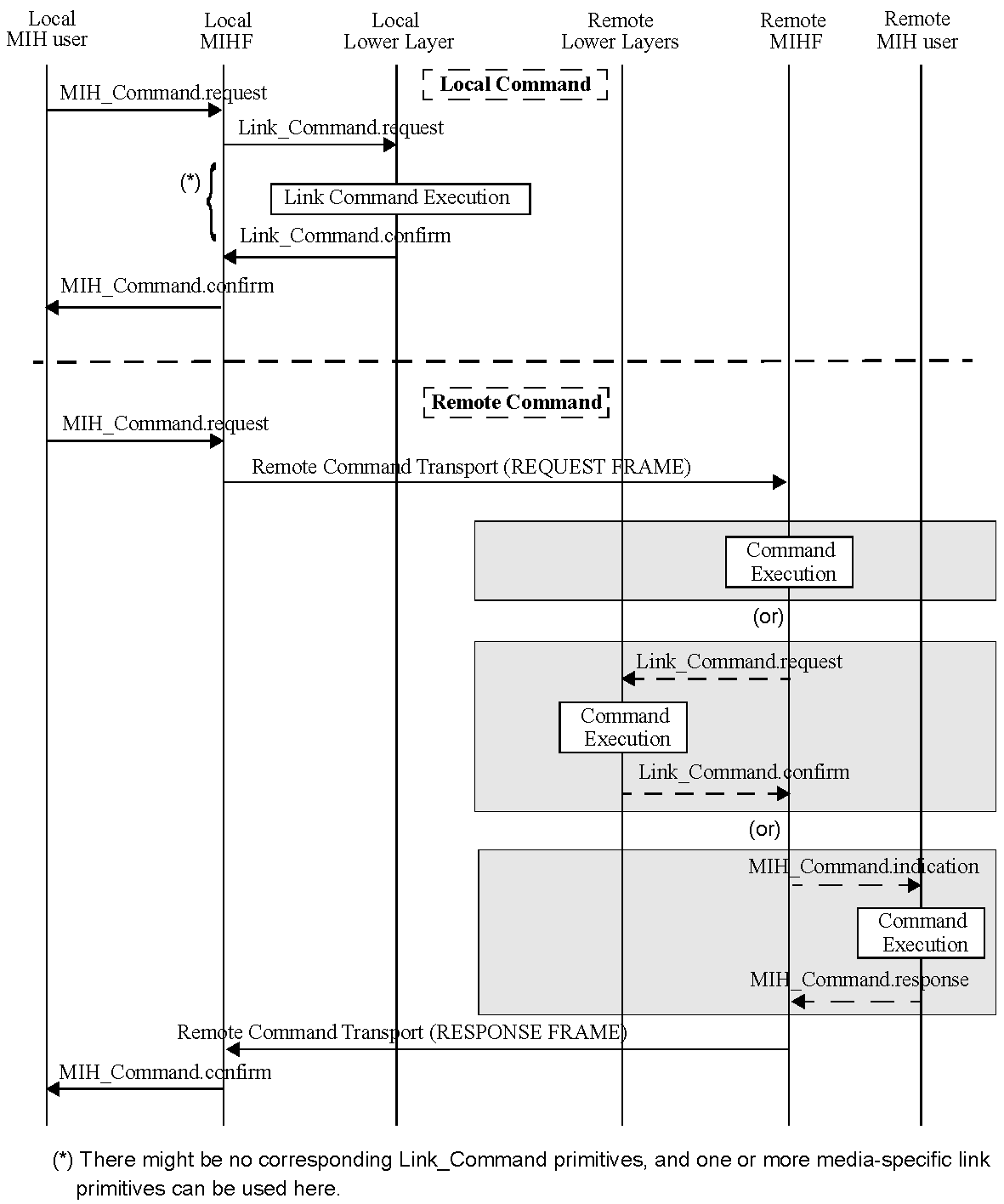


1. —Remote MIS command

Command service flow model

The MIS commands are generated by the MIS users and sent to the MISF. MIS commands can be local or remote. Local MIS commands are sent by MIS users to the MISF in the local protocol stack. Generally, remote commands generate an appropriate response frame from a remote MIS user, however, there are certain remote commands that do not (cf. downlink-only technology related MIS commands).

Figure 17 shows the flow for a local command and an example of a remote command, respectively. Example handover procedures using the commands defined in 6.4.3 can be found in Annex C. Remote commands are transported over network layer protocols or link-layer protocols.



1. —Command service flow

Command list

* + - * 1. Link commands

Table 6 defines Link commands.

1. —Link commands

|  |  |  |
| --- | --- | --- |
| **Link command** | **Comments** | **Definedin** |
| Link\_Capability\_Discover | Query and discover the list of supported link-layer events and link- layer commands. | 7.3.9 |
| Link\_Event\_Subscribe | Subscribe to one or more events from a link. | 7.3.10 |
| Link\_Event\_Unsubscribe | Unsubscribe from a set of link-layer events. | 7.3.11 |
| Link\_Get\_Parameters | Get parameters measured by the active link, such as signal-to-noise ratio (SNR), BER, received signal strength indication (RSSI). | 7.3.12 |
| Link\_Configure\_Thresholds | Configure thresholds for Link Parameters Report event. | 7.3.13 |
| Link\_Action | Request an action on a link-layer connection. | 7.3.14 |

* + - * 1. MIS commands

General

Table 7 defines MIS Commands. An MIS command is marked as local only (L), remote only (R), or local and remote (L, R), indicating whether it can be issued by a local MIS user, a remote MIS user, or both, respectively.

1. —MIS commands

|  |  |  |  |
| --- | --- | --- | --- |
| **MIS command** | **(L) ocal, (R) emote** | **Comments** | **Defined in** |
| MIS\_Link\_Get\_Parameters | L, R | Get the status of a link. | 7.4.14 |
| MIS\_Link\_Configure\_Thresholds | L, R | Configure link parameter thresholds. | 7.4.15 |
| MIS\_Link\_Actions | L, R | Control the behavior of a set of links. | 7.4.16 |
| MIS\_Net\_HO\_Candidate\_Query | R | Network initiates handover and sends a list of suggested networks and associated points of attachment. | 7.4.17 |
| MIS\_MN\_HO\_Candidate\_Query | R | Command used by MN to query and obtain handover related information about possible candidate networks. | 7.4.18 |
| MIS\_N2N\_HO\_Query\_Resources | R | This command is sent by the serving MISF entity to the target MISF entity to allow for resource query. | 7.4.19 |
| MIS\_MN\_HO\_Commit | R | Command used by MN to notify the serving net- work of the decided target network information. | 7.4.20 |
| MIS\_Net\_HO\_Commit | R | Command used by the network to notify the MN of the decided target network information. | 7.4.2 1 |
| MIS\_N2N\_HO\_Commit | R | Command used by a serving network to inform a target network that an MN is about to move toward that network, initiate context transfer (if applicable), and perform handover preparation. | 7.4.22 |
| MIS\_MN\_HO\_Complete | R | Notification from MISF of the MN to the target or source MISF indicating the status of handover completion. | 7.4.23 |
| MIS\_N2N\_HO\_Complete | R | Notification from either source or target MISF to the other (i.e., peer) MISF indicating the status of the handover completion. | 7.4.24 |

Naming convention for MIS handover commands (to be excluded)

Mobile initiated handovers (to be excluded)

Network initiated handovers (to be excluded)

* 1. Media independent information service

Introduction

Media independent information service (MIIS) provides a framework by which an MISF, residing in the MN or in the network, discovers and obtain network information within a geographical area to facilitate network selection and handovers. The objective is to acquire a global view of all the heterogeneous networks relevant to the MN in the area to facilitate seamless roaming across these networks.

MIIS includes support for various Information Elements (IEs). IEs provide information that is essential for a network selector to make intelligent handover decisions.

Depending on the type of mobility, support for different types of information elements is required for performing handovers. MIIS provides the capability for obtaining information about lower layers such as neighbor maps and other link-layer parameters, as well as information about available higher layer services such as internet connectivity.

MIIS provides a generic mechanism to allow a service provider and a mobile user to exchange information on different handover candidate access networks. The handover candidate information includes different access technologies such as IEEE 802 networks, 3GPP networks, and 3GPP2 networks. The MIIS also allows this collective information to be accessed from any single network. For example, by using an IEEE 802.11 access network the MN gets information not only about all other IEEE 802 based networks in a particular region but also about 3GPP and 3GPP2 networks. Similarly by using a 3GPP2 interface, the MN gets access to information about all IEEE 802 and 3GPP networks in a given region. This capability allows the MN to use its currently active access network and inquire about other available access networks in a geographical region. Thus, an MN is freed from the burden of powering up each of its individual radios and establishing network connectivity for the purpose of retrieving heterogeneous network information. MIIS enables this functionality across all available access networks by providing a uniform way to retrieve heterogeneous network information in any geographical area.

The main goal behind the Information Service is to allow MN and network entities to discover information that influences the selection of appropriate networks during handovers. This information is intended to be primarily used by a policy engine entity that can make effective handover decisions based on this information. This Information Service provides mostly static information, although network configuration changes are also accounted for. Other dynamic information about different access networks, such as current available resource levels, state parameters, and dynamic statistics should be obtained directly from the respective access networks. Some of the key motivations behind the Information Service are as follows:

1. Provide information about the availability of access networks in a geographical area. Further, this information could be retrieved using any wireless network, for example, information about a nearby Wi-Fi hotspot could be obtained using a global system for mobile communication (GSM), CDMA, or any other cellular network, whether by means of request/response signaling, or by means of information that is specifically or implicitly broadcast over those cellular networks. Alternatively, this information could be maintained in an internal database on the MN.
2. Provide static link-layer information parameters that help the MNs in selecting the appropriate access network. For example knowledge of whether security and QoS are supported on a particular access network influences the decision to select such an access network during handovers.
3. Provide information about capabilities of different PoAs in neighbor reports to aid in configuring the radios optimally (to the extent possible) for connecting to available or selected access networks. For example knowing about supported channels by different PoAs helps in configuring the channels optimally as opposed to scanning or beaconing and then finding out this information. Dynamic link- layer parameters have to be obtained or selected based on direct interaction with the access networks.
4. Provide an indication of higher layer services supported by different access networks and core networks that can aid in making handover decisions. Such information is not available directly from the MAC sublayer or PHY of specific access networks, but can be provided as part of the Information Service. For example, classification of different networks into categories, such as public, enterprise, home, and others, influences a handover decision. These higher layer services information is more vendor specific in nature.

Access information service before authentication

It is important to note that, with certain access networks an MN should be able to obtain IEEE 802.21 related information elements before the MN is authenticated with the PoA. These information elements are used by the handover policy function to determine if the PoA can be selected. In order to enable the information query before authentication, individual link technologies provide an L2 or media-specific transport or a protocol message exchange that makes this MIIS query exchange possible between the user equipment (MN) and a certain MISF in the network. It should be noted that the pre-authentication query facility is provided only for MIS information query and cannot be used for carrying other MIS protocol services except MISF capability discovery query using MIS\_Capability\_Discover embedded into media specific management frames. Additionally, any MISF within the network can request the set of information elements from a peer MISF located in the same or a different network using the MIS protocol.

Allowing access of information service before authentication carries certain security risks such as denial-of­service attacks and exposure of information to unauthorized MNs. In such scenarios the information service provider limits the scope of information accessible to an unauthenticated MN.

After authentication and attachment to a certain PoA, the MIS protocol is used for information retrieval by use of data frames specific to that media technology.

Restricting query response size

When sending an information query request, the MIIS client provides a maximum response size to limit the query response message size. A request can contain multiple queries. If the request contains multiple queries, they will be in the order of significance to the client. In case the query results exceed the maximum response size, the least significant query results will be removed from the response. The MIIS server has its own maximum response size limit configured that is smaller than the one specified by the MIIS client request. In this case, the response message returns results in the order of significance to the client up to that limit.

Information elements

The Information Service elements are classified into the following three groups:

1. General Information and Access Network Specific Information: These information elements give a general overview of the different networks providing coverage within an area. For example, a list of available networks and their associated operators, roaming agreements between different operators, cost of connecting to the network and network security and quality of service capabilities.
2. PoA Specific Information: These information elements provide information about different PoAs for each of the available access networks. These IEs include PoA addressing information, PoA location, data rates supported, the type of PHY and MAC layers and any channel parameters to optimize link-layer connectivity. This also includes higher layer services and individual capabilities of different PoAs.
3. Other information that is access network specific, service specific, or vendor/network specific.

Table 9 lists information element containers (see 6.5.6.2.1 for detailed definitions). The containers are only used in the type-length-value (TLV) based query method.

1. —Information element containers

|  |  |
| --- | --- |
| **Name of container** | **Description** |
| IE\_CONTAINER\_LIST\_OF\_NETWORKS | List of neighboring Access Network Containers, containing information that depicts a list of heterogeneous neighboring access networks for a given geographical location. |
| IE\_CONTAINER\_NETWORK | Access Network Container, containing information that depicts an access network. |
| IE\_CONTAINER\_POA | PoA Container, containing information that depicts a PoA. |

Table 10 represents the list of Information Elements and their semantics. Each Information Element has an abstract data type (see Annex F for detailed definitions). The binary and resource description framework (RDF) representation of these Information Elements are described in 6.5.6.2 and 6.5.6.3, respectively. The IEs may be retrieved using TLV or SPARQL based query methods. The standard does not recommend or mandate the choice of either method. An IEEE 802.21 implementation that implements the MIIS shall implement at least one method. Vendors or network operators define additional IEs beyond the IEs specified in Table 10. Vendors and network operators can implement new IEs using the Vendor Specific IEs. These IEs will then be available only in vendor- or operator-specific deployments.

1. —Information elements

|  |  |  |
| --- | --- | --- |
| **Name of information element** | **Description** | **Data type** |
| **General information elements** | | |
| IE\_NETWORK\_TYPE | Link types of the access networks that are available in a given geographical area. | NETWORK\_TYPE |
| IE\_OPERATOR\_ID | The operator identifier for the access network/core network. | OPERATOR\_ID |
| IE\_SERVICE\_PROVIDER\_ID | Identifier for the service provider. | SP\_ID |
| IE\_COUNTRY\_CODE | Indicate the country. | CNTRY\_CODE |

Table 10—Information elements (continued)

|  |  |  |  |
| --- | --- | --- | --- |
| Name of information element | Description | | Data type |
| IE\_SERVICE\_PROVIDER\_ID | Identifier for the service provider. | | SP\_ID |
| IE\_COUNTRY\_CODE | Indicate the country. | | CNTRY\_CODE |
| Access network specific information elements | | | |
| IE\_NETWORK\_ID | | Identifier for the access network. | NETWORK\_ID |
| IE\_NETWORK\_AUX\_ID | | An auxiliary access network identifier. As an example for IEEE 802.11 this refers to the homogenous extended service set ID (HESSID). | NET\_AUX\_ID |
| IE\_ROAMING\_PARTNERS | | Roaming Partners.  Network Operators with which the current network operator has direct roaming agreements. | ROAMING\_PTNS |
| IE\_COST | | Cost.Indication of cost for service or network usage. | COST |
| IE\_NETWORK\_QOS | | QoS characteristics of the link layer. | QOS\_LIST |
| IE\_NETWORK\_DATA\_RATE | | Data Rate. The maximum value of the data rate sup- ported by the link layer of the access network. | DATA\_RATE |
| IE\_NET\_REGULAT\_DOMAIN | | Regulatory classes supported by the access network. | REGU\_DOMAIN |
| IE\_NET\_FREQUENCY\_BANDS | | Frequency bands supported by the network. | FREQ\_BANDS |
| IE\_NET\_IP\_CFG\_METHODS | | IP Configuration Methods supported by the access network. | IP\_CONFIG |
| IE\_NET\_CAPABILITIES | | Bitmap of access network capabilities. | NET\_CAPS |
| IE\_NET\_SUPPORTED\_LCP | | List of location configuration protocols supported by the access network. | SUPPORTED\_LCP |
| IE\_NET\_MOB\_MGMT\_PROT | | Type of mobility management protocol supported. | IP\_MOB\_MGMT |
| IE\_NET\_EMSERV\_PROXY | | Address of the proxy providing access to public safety answering point (PSAP). | PROXY\_ADDR |
| IE\_NET\_IMS\_PROXY\_CSCF | | Address of the proxy providing access to IMS P-CSCF. | PROXY\_ADDR |
| IE\_NET\_MOBILE\_NETWORK | | Indicator whether the access network itself is mobile. | BOOLEAN |
| **PoA-specific information elements** | | | |
| IE\_POA\_LINK\_ADDR | | Link-layer address of PoA. | LINK\_ADDR |
| IE\_POA\_LOCATION | | Geographical location of PoA. Multiple location types are supported including coordinate-based location information, civic address, and cell ID. | LOCATION |
| IE\_POA\_CHANNEL\_RANGE | | Channel Range/Parameters.  Spectrum range supported by the channel for that PoA. | CH\_RANGE |
| IE\_POA\_SYSTEM\_INFO | | System information supported by the link layer of a given PoA. | SYSTEM\_INFO |
| IE\_AUTHENTICATOR\_LINK\_ADDR | | An L2 address of the authenticator, which serves the PoA | LINK\_ADDR |
| **PoA-specific higher layer service information elements** | | | |

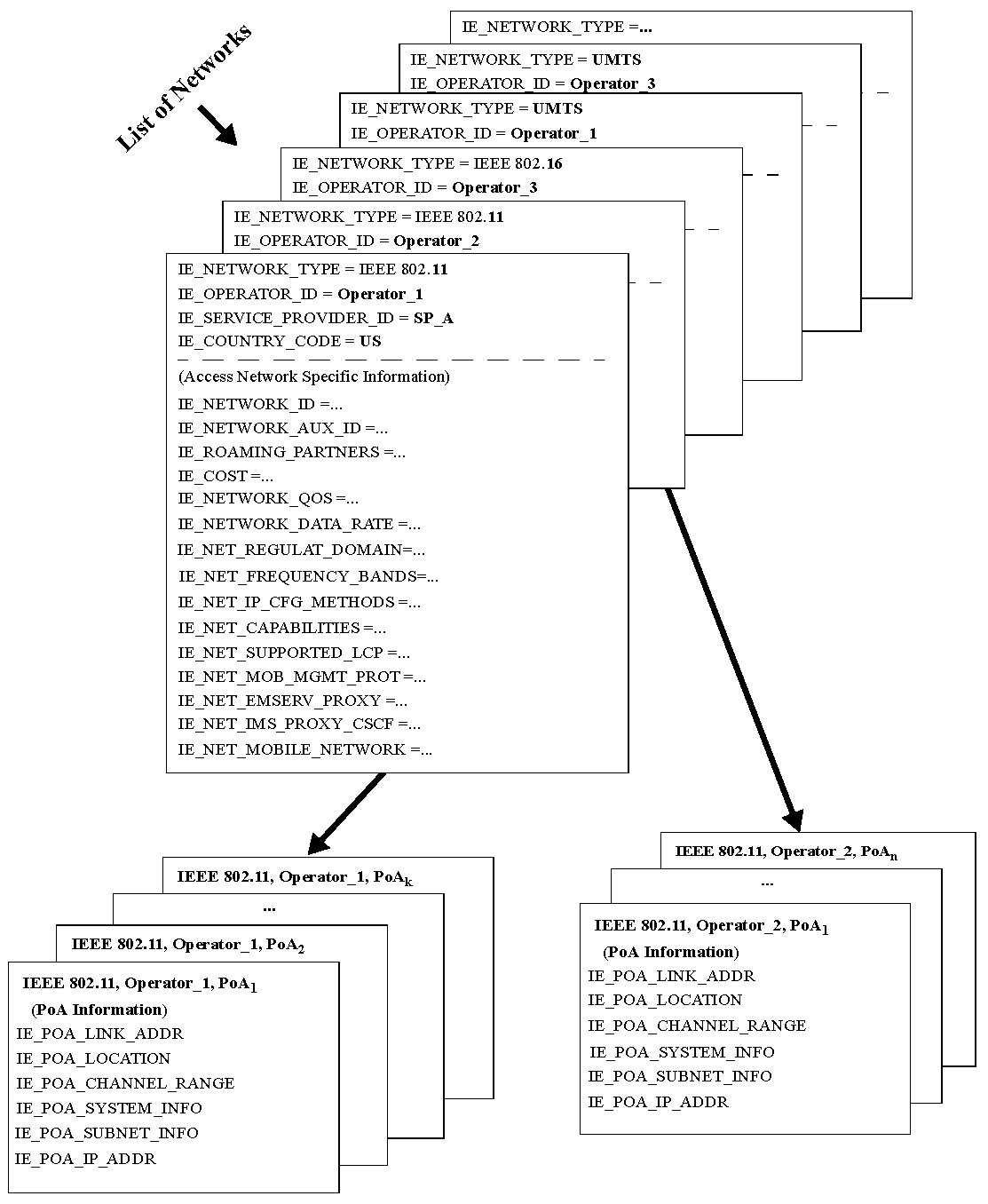
**Table 10—Information elements *(continued)***

|  |  |  |
| --- | --- | --- |
| **Name of information element** | **Description** | **Data type** |
| IE\_POA\_SUBNET\_INFO | Information about subnets supported by a typical PoA. | IP\_SUBNET\_INFO |
| IE\_POA\_IP\_ADDR | IP Address of PoA. | IP\_ADDR |
| IE\_AUTHENTICATOR\_IP\_ADDR | The IP address of the authenticator, which serves the PoA. | IP\_ADDR |
| IE\_PoS\_IP\_ADDR | PoS’s IP address. | IP\_ADDR |
| **Other information elements** | | |
| Vendor specific IEs | Vendor-specific services. | N/A |

In certain access network deployments, some PoA properties (e.g., data rate, IP configuration methods, capabilities) are common for all PoAs within that access network. In such a case, the common PoA properties are represented as IEs as part of the access network property information.

As an example, Figure 18 shows the layout of different Information Elements and the neighbor map of different networks in a geographical area. Multiple operators can be providing support for a particular network. Thus support for IEEE 802.11 network is provided by both Operator\_1 and Operator\_2. A single operator can provide support for multiple networks. Thus, Operator\_1 provides support for IEEE 802.11 and universal mobile telecommunications system (UMTS) networks while Operator\_3 provides support for IEEE 802.16 and UMTS networks. The General Network Information Elements are specified for each network supported by an operator. Thus in the case of Operator\_1, General Network Information is specified for both IEEE 802.11 and UMTS networks, while in the case of Operator\_2 it is specified only for an IEEE 802.11 network.

For each network supported by an operator there is a list of supported PoAs. For each PoA the PoA Information Elements are specified. Figure 18 shows this information representation and tree hierarchy for different networks.



1. —Depicting a list of neighboring networks with information elements

Definition of information element namespace

Each Information Element ID is a 32 bit value. Table 11 defines the Information Element namespace. The IEEE 802.21 specific Information Elements are assigned identifiers as per this standard. Please refer to Table G. 1 (in Annex G) for more details. Vendors specify their own IEs using the name space allocated to them. A set of IE name space ranges is also reserved for development and testing. These should not be used in released products. Allocation of additional IE namespace and any revisions to this assignment will be handled by future revisions of this standard.

1. —Information element namespace

|  |  |  |
| --- | --- | --- |
| **Range** | **Description** | **Comments** |
| 0x00000000 | Reserved |  |
| 0x00000001–0x1FFFFFFF | IEEE 802.21 IEs | Used for IEEE 802.21 defined IEs. The cur­rently defined IEEE 802.21 IEs are listed in Table G.1 in Annex G. |
| 0x20000000–0x7FFFFFFF | Vendor specific IEs | Used for IEs defined by vendors.  To prevent vendor specific IE collisions, the 2nd, 3rd, and 4th octet are filled with the value of the vendor’s IEEE organizationally unique identifier (OUI). For example, if a vendor’s IEEE OUI is 00-03-3F, then its corresponding Vendor Specific IE range would be 0x2000033F–0x7F00033F. |
| 0x80000000–0x82FFFFFF | Reserved for playpen area. | Used in development and testing. Should not be used in released products. Avoids collision during development. |
| 0x83000000–0xFFFFFFFF | Reserved | For future use. |

Functional entities should discard any received IE with an unrecognizable identifier.

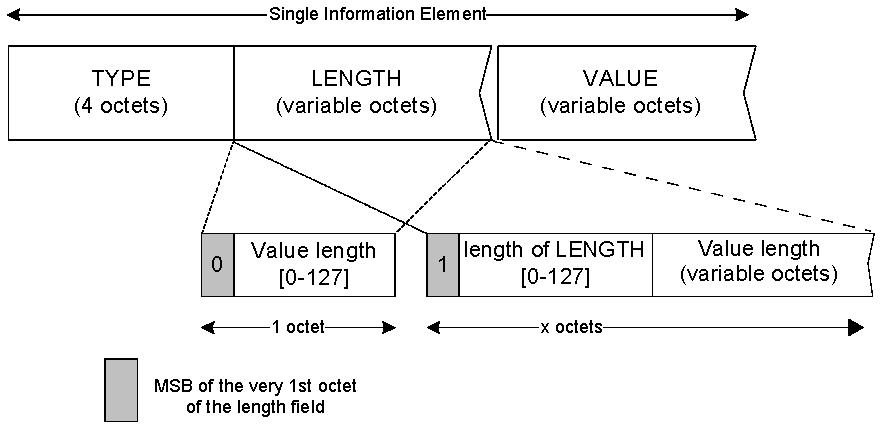
Information element representation and query methods

* + - * 1. Introduction

MIIS defines two methods for representing Information Elements: binary representation and RDF representation (see W3C Recommendation, Resource Description Framework (RDF)–Concepts and Abstract Syntax and W3C Recommendation, RDF/XML Syntax Specification). MIIS also defines two query methods. For requests using the binary representation, the TLV query method defined in 6.5.6.2 is used. For requests using the RDF representation, the SPARQL (see W3C Recommendation, SPARQL Query Language for RDF) query method is used.

* + - * 1. Binary representation and TLV query

In the binary representation method, Information Elements are represented and encoded in Type-Length­Value form as shown in Figure 19.



1. —TLV representation of information elements

The *Length* field is interpreted as follows:

Case 1: If the number of octets occupied by the *Value* field is less than 128, the size of the *Length* field is always one octet and the MSB of the octet is set to the value ‘0’. The values of the other seven bits of this octet indicate the actual length of the *Value* field.

Case 2: If the number of octets occupied by the *Value* field is exactly 128, the size of the *Length* field is one octet. The MSB of the *Length* octet is set to the value ‘1’ and the other seven bits of this octet are all set to the value ‘0’.

Case 3: If the number of octets occupied by the *Value* field is greater than 128, then the *Length* field is always greater than one octet. The MSB of the first octet of the *Length* field is set to the value ‘1’ and the remaining seven bits of the first octet indicate the number of octets that are appended further. The number represented by the second and subsequent octets of the *Length* field, when added to 128, indicates the total size of the *Value* field, in octets.

IE containers

In the binary representation method, three Information Element Containers are defined, namely the IE\_CONTAINER\_LIST\_OF\_NETWORKS, the IE\_CONTAINER\_NETWORK, and the IE\_CONTAINER\_POA:

* **IE\_CONTAINER\_LIST\_OF\_NETWORKS**—contains a list of heterogeneous neighboring access networks for a given geographical location, as shown in Table 12.

An IE\_CONTAINER\_LIST\_OF\_NETWORKS contains at least one Access Network and option­ally one or more Vendor Specific IEs. When more than one Access Network Container is provided in this IE, they should be prioritized in the order of preference from the information server’s per­spective with first Access Network Container as the top priority and with decreasing priority going down the list. This would enable the receiving entity to utilize this information in the same way as provided in this list for network selection or handover decisions.

1. —IE\_CONTAINER\_LIST\_OF\_NETWORKS definition

|  |  |
| --- | --- |
| **Information element ID =** (see Table G.1) | **Length=** *variable* |
| IE\_CONTAINER\_NETWORK #1 | |
| IE\_CONTAINER\_NETWORK #2 (optional) | |
| … | |
| IE\_CONTAINER\_NETWORK #k (optional) | |
| Vendor Specific IE (optional) | |

* **IE\_CONTAINER\_NETWORK**—contains all the information depicting an access network, as shown in Table 13.

When more than one PoA Container is provided in this IE, they should be prioritized in the order of preference from the information server’s perspective with first PoA Container as the top priority and with decreasing priority going down the list. This would enable the receiving entity to utilize this information in the same way as provided in this list for network selection or handover decisions.

1. —IE\_CONTAINER\_NETWORK definition

|  |  |
| --- | --- |
| **Information element ID =** (see Table G.1) | **Length =** *variable* |
| IE\_NETWORK\_TYPE | |
| IE\_OPERATOR\_ID | |
| IE\_SERVICE\_PROVIDER\_ID (optional) | |
| IE\_COUNTRY\_CODE (optional) | |
| IE\_NETWORK\_ID (optional) | |
| IE\_NETWORK\_AUX\_ID (optional) | |
| IE\_ROAMING\_PARTNERS (optional) | |
| IE\_COST (optional) | |
| IE\_NETWORK\_QOS (optional) | |
| IE\_NETWORK\_DATA\_RATE (optional) | |
| IE\_NET\_REGULAT\_DOMAIN (optional) | |
| IE\_NET\_FREQUENCY\_BANDS (optional) | |
| IE\_NET\_IP\_CFG\_METHODS (optional) | |
| IE\_NET\_CAPABILITIES (optional) | |
| IE\_NET\_SUPPORTED\_LCP (optional) | |
| IE\_NET\_MOB\_MGMT\_PROT (optional) | |
| IE\_NET\_EMSERV\_PROXY (optional) | |
| IE\_NET\_IMS\_PROXY\_CSCF (optional) | |

**Table 13—IE\_CONTAINER\_NETWORK definition *(continued)***

|  |
| --- |
| IE\_NET\_MOBILE\_NETWORK (optional) |
| IE\_CONTAINER\_POA #1 (optional) |
| IE\_CONTAINER\_POA #2 (optional) |
| **…** |
| IE\_CONTAINER\_POA #k (optional) |
| Vendor Specific Network IE (optional) |

* **IE\_CONTAINER\_POA**—contains all the information depicting a PoA and optionally one or more Vendor Specific PoA IEs, as shown in Table 14.

1. —IE\_CONTAINER\_POA definition

|  |  |
| --- | --- |
| **Information element ID =** (see Table G.1) | **Length=** *variable* |
| IE\_POA\_LINK\_ADDR | |
| IE\_POA\_LOCATION | |
| IE\_POA\_CHANNEL\_RANGE | |
| IE\_POA\_SYSTEM\_INFO | |
| IE\_POA\_SUBNET\_INFO #1 | |
| IE\_POA\_SUBNET\_INFO #2 (optional) | |
| ... | |
| IE\_POA\_SUBNET\_INFO #k (optional) | |
| IE\_POA\_IP\_ADDR #1 (optional) | |
| ... | |
| IE\_POA\_IP\_ADDR #k (optional) | |
| Vendor Specific PoA IE (optional) | |

TLVs for the component IEs contained in the Access Network Container and PoA Container are defined in Annex F.

TLV queries

A TLV query includes the following optional parameters to refine the query.

QUERIER\_LOC parameter (defined in Table F. 15) can be useful for the Information Server to refine its response. The value field contains either the Querier's current location measurement or, when the querier does not have its current location information, an observed link-layer address (e.g., from an IEEE 802.11 Beacon frame or some broadcast mechanism for other technologies) that the Information Server will be able to use as a hint to establish an estimate of the client’s current location. Within the QUERIER\_LOC parameter, the querier should not use both the LINK\_ADDR value (defined in Table F.3) and LOCATION value (defined in Table F.10) in the same query. Moreover, the NGHB\_RADIUS value (defined in Table F. 15), if provided, indicates the radius of the neighborhood, centered at the indicated location, within which all available access networks will be included in the list of neighboring networks. If NGHB\_RADIUS value is not present, the Information Server will decide the radius for the search.

If QUERIER\_LOC parameter is not included in the query, the Information Server either gets the querier location information through other means or uses the best estimate of the querier’s location to generate the neighboring network information.

NET\_TYPE\_INC parameter (see Table F. 15 for definition) can be used to indicate the neighboring network types the querier wants to include in the response. The querier indicates the network types it wants to include in the query response by setting the corresponding bits to “1”. If not provided, the Information Server includes information about all available network types in the query response.

NETWK\_INC parameter (see Table F. 15 for definition) can be used to indicate the specific access networks the querier wants to include in the query response. If not provided, the Information Server includes information about all available access networks in the query response.

RPT\_TEMPL parameter (see Table F. 15 for definition) can be used to give the information server a template of the list of IEs that is included in the information response.

The following rules shall be followed for using RPT\_TEMPL parameter:

* If the RPT\_TEMPL parameter is absent, the entire list of neighboring networks container is returned in the response (subject to constraints on message length, as defined in 6.5.3).
* If a container is listed *without* any of its component IEs, the entire container is returned in the response (subject to constraints on message length, as defined in 6.5.3). For example, inclusion of *IE\_CONTAINER\_POA* solely returns a list of PoA Containers with all their component IEs.
* If a container is listed *with* one or more of its component IEs, the container *with only* the listed component IEs is returned. For example, inclusion of *IE\_CONTAINER\_NETWORK, IE\_NETWORK\_TYPE* and *IE\_OPERATOR\_ID* solely returns a list of Network Containers with each containing only Network Type and Operator ID.
* If a component IE is listed *without* its parent container, the listed component IE is returned as an individual IE. For example, inclusion of *IE\_NETWORK\_TYPE and IE\_COST* solely returns a list of Network Types and a list of Costs.

NOTE—A list of individual IEs out of their context has very limited usefulness. This is only an example to show the flexible use of RPT\_TEMPL parameter.[[13]](#footnote-13)

The following rules are followed for generating returned IEs:

Upon receipt of a binary query, the information server will

1. Create the list of neighboring access network information for the given location.
   1. If a NET\_TYPE\_INC parameter is provided in the query, include only the information of the neighboring access networks of the network type(s) indicated in the NET\_TYPE\_INC parameter. Otherwise, include information of all available neighboring access networks for the given location.
   2. If a NETWK\_INC parameter is provided in the query, include only the information of the neighboring access network(s) indicated in the NETWK\_INC parameter. Otherwise, include information of all available neighboring access networks for the given location.
2. If no RPT\_TEMPL parameter is given in the query, send the list of neighboring access network information in an IE\_CONTAINER\_LIST\_OF\_NETWORKS in an MIS\_Get\_Information response message.
3. If an RPT\_TEMPL parameter is given in the query, extract the requested IE(s)/Containers from the list of neighboring access network information using the rules described for RPT\_TEMPL parameter and send them in an MIS\_Get\_Information response message.
   * + - 1. RDF representation and SPARQL query

The RDF representation of Information Elements is represented in XML format. SPARQL is used as the query method. The RDF representation and SPARQL query method will implement the RDF schema as described in 6.5.7.2.

Information service schema

* + - * 1. General

A schema is used in the IEEE 802.21 Information Service to define the structure of each information element, as well as the relationship among the information elements. The IEEE 802.21 Information Service schema is supported by every MISF that implements the MIIS to support flexible and efficient information queries.

* + - * 1. MIIS RDF schema

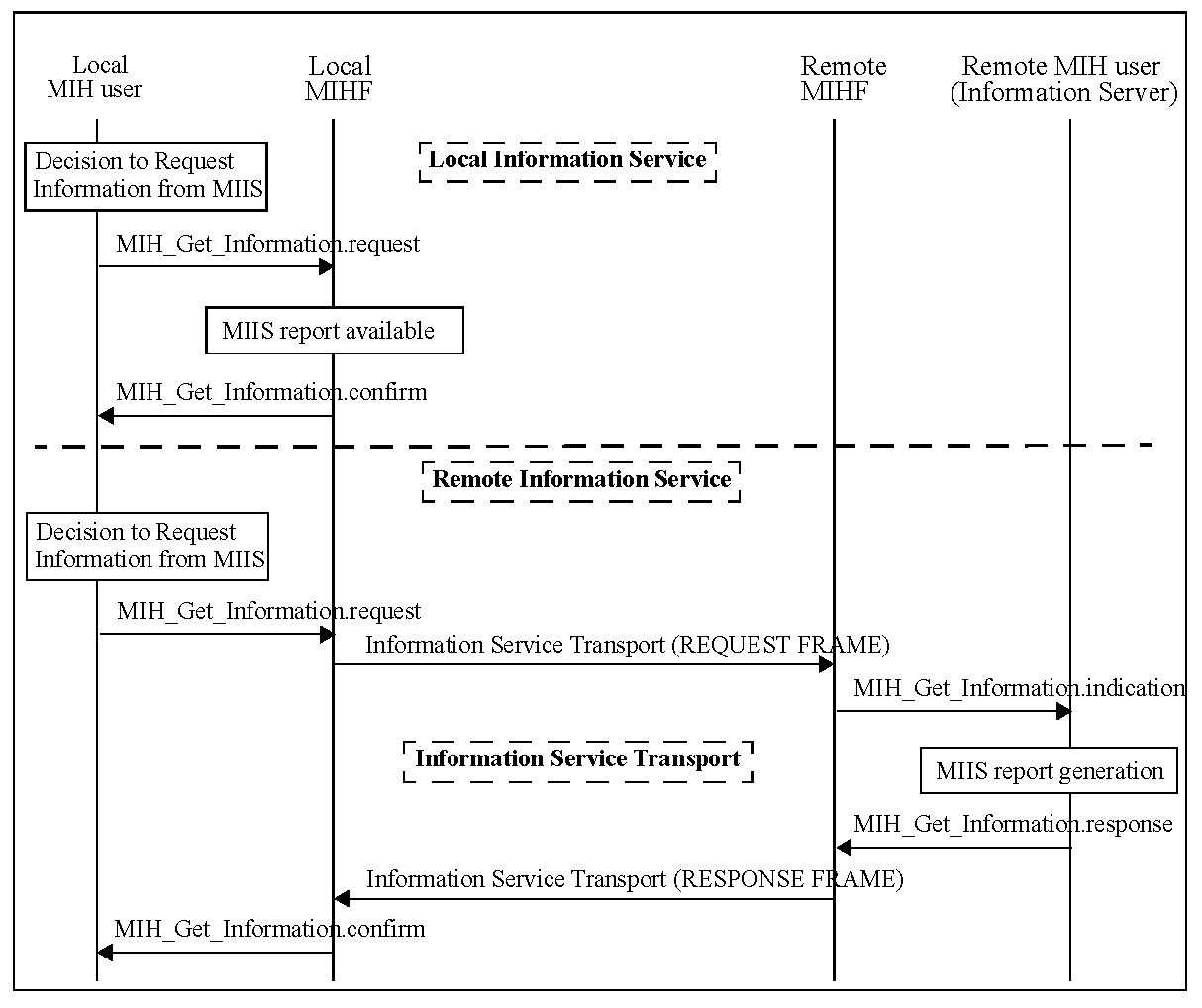
The RDF schema definition for MIIS consists of two parts; the basic and the extended schema. An MIIS client or server should be pre-provisioned with the basic schema for ease of implementation of schema- based query. In scenarios where the basic schema is not pre-provisioned, methods such as dynamic host configuration protocol (DHCP) are used to obtain the basic schema.

The MIIS RDF representation method is extensible using the extended schema. The extended schema can be pre-provisioned. The extended schema can also be updated dynamically, e.g., when a new information element about the network is introduced. When the extended schema is not pre-provisioned it is retrieved from the specified URL via the IEEE 802.21 Information Service using the schema query capability. Alternatively, methods such as DHCP provide the URL of the extended schema as well. The implementation will always use the updated version of extended schema as opposed to using the pre-provisioned version.

The basic schema is defined in Annex H. The basic schema contains the schema for information elements defined in Table 10. The extended schema is defined by individual vendors or by network operators and contains the schema for vendor-specific information elements or network operator specific information. (See Annex I for an example of a vendor-specific extension.)

Information service flow

Figure 20 describes an Information Service flow. The MIIS within an MISF communicates with the remote MISF that resides within the access network. MIS\_Get\_Information from the MN is carried over the appropriate transport (L2 or L3) and is delivered to the remote MISF. The remote MISF returns the necessary information to the MN via the appropriate response frame.



**Figure 20—MIIS information flow**

1. Service access points (SAPs) and primitives
   1. Introduction

The MIS Function uses the following SAPs for interfacing with other entities.

**Media dependent SAPs:**

1. MIS\_LINK\_SAP: Abstract media dependent interface of MISF with the lower layers of the media- specific protocol stacks. The mappings between MIS\_LINK\_SAP and various media-specific SAPs are described in E.2.
2. MIS\_NET\_SAP: Abstract media dependent interface of MISF that provides transport services over the data plane on the local node, supporting the exchange of MIS information and messages with the remote MISF.

**Media independent SAP:**

* MIS\_SAP: This SAP defines the media independent interface between the MISF and MIS users.
  1. SAPs

General

The SAPs are defined as a set of primitives. Taken together, the primitives define the services. Within the definition of each primitive there is a table of allowable parameters. Each parameter is defined using abstract data types. These types indicate the semantic value of that parameter. The parameters defined within the subclause for a particular primitive are produced or consumed by that primitive. Several of the abstract data types are used in multiple primitive definitions. In each abstract data type definition, the various names applied to this type are listed in Annex F.

Media dependent SAPs

* + - * 1. MIS\_LINK\_SAP

The primitives defined as part of the MIS\_LINK\_SAP are described in Table 15. Annex E contains their mapping to several specific link technologies. IETF RFC 5184 [B27] specifies many of these primitives as L2 abstractions.

1. —MIS\_LINK\_SAP primitives

|  |  |  |  |
| --- | --- | --- | --- |
| **Primitives** | **Service category** | **Description** | **Defined in** |
| Link\_Detected | Event | A new link is detected | 7.3.1 |
| Link\_Up | Event | L2 connectivity is established | 7.3.2 |
| Link\_Down | Event | L2 connectivity is lost | 7.3.3 |
| Link\_Parameters\_Report | Event | Link parameters have crossed specified thresholds | 7.3.4 |
| Link\_Going\_Down | Event | L2 connectivity loss is imminent | 7.3.5 |
| Link\_Handover\_Imminent | Event | L2 handover is imminent | 7.3.6 |

**Table 15—MIS\_LINK\_SAP primitives *(continued)***

|  |  |  |  |
| --- | --- | --- | --- |
| **Primitives** | **Service category** | **Description** | **Defined in** |
| Link\_Handover\_Complete | Event | L2 handover has been completed | 7.3.7 |
| Link\_PDU\_Transmit\_Status | Event | Indicate transmission status of a PDU | 7.3.8 |
| Link\_Capability\_Discover | Command | Query and discover the list of supported link-layer events and link-layer commands | 7.3.9 |
| Link\_Event\_Subscribe | Command | Subscribe for event notifications | 7.3.10 |
| Link\_Event\_Unsubscribe | Command | Unsubscribe from event notifications | 7.3.11 |
| Link\_Get\_Parameters | Command | Request parameters of medium | 7.3.12 |
| Link\_Configure\_Thresholds | Command | Configure link thresholds for Link events | 7.3.13 |
| Link\_Action | Command | Request an action on a link-layer connection | 7.3.14 |

* + - * 1. MIS\_NET\_SAP

The primitive defined for MIS\_NET\_SAP is described in Table 16.

1. —MIS\_NET\_SAP primitive

|  |  |  |  |
| --- | --- | --- | --- |
| **Primitive** | **Service category** | **Description** | **Defined in** |
| MIS\_TP\_Data | Network communication | This primitive is used for transfer of data | 7.5.1 |

Media independent SAP: MIS\_SAP (some primitives deleted)

The primitives defined as part of MIS\_SAP are described in Table 17.

1. —MIS\_SAP primitives

|  |  |  |  |
| --- | --- | --- | --- |
| **Primitives** | **Service category** | **Description** | **Defined in** |
| MIS\_Capability\_Discover | Service management | Discover list of Events and Commands sup- ported by MISF | 7.4.1 |
| MIS\_Register | Service management | Register with a remote MISF | 7.4.2 |
| MIS\_DeRegister | Service management | Deregister with a remote MISF | 7.4.3 |
| MIS\_Event\_Subscribe | Service management | Subscribe for MIS event notifications | 7.4.4 |
| MIS\_Event\_Unsubscribe | Service management | Unsubscribe from MIS event notifications | 7.4.5 |
| MIS\_Link\_Detected | Event | A new link is detected | 7.4.6 |

**Table 17—MIS\_SAP primitives *(continued)***

|  |  |  |  |
| --- | --- | --- | --- |
| **Primitives** | **Service category** | **Description** | **Defined in** |
| MIS\_Link\_Up | Event | L2 connection has been established | 7.4.7 |
| MIS\_Link\_Down | Event | L2 connectivity is lost | 7.4.8 |
| MIS\_Link\_Parameters\_Report | Event | Link parameters have crossed specified threshold | 7.4.9 |
| MIS\_Link\_Going\_Down | Event | L2 connectivity is predicted to go down | 7.4.10 |
| MIS\_Link\_Handover\_Imminent | Event | L2 handover is imminent | 7.4.11 |
| MIS\_Link\_Handover\_Complete | Event | L2 handover has been completed | 7.4.12 |
| MIS\_Link\_PDU\_Transmit\_Status | Event | Indicate transmission status of a PDU | 7.4.13 |
| MIS\_Link\_Get\_Parameters | Command | Get the status of link | 7.4.14 |
| MIS\_Link\_Configure\_Thresholds | Command | Configure link parameter thresholds | 7.4.15 |
| MIS\_Link\_Actions | Command | Control the behavior of a set of links | 7.4.16 |
| MIS\_Net\_HO\_Candidate\_Query | Command | Initiate handover | 7.4.17 |
| MIS\_MN\_HO\_Candidate\_Query | Command | Initiate MN query request for candidate network | 7.4.18 |
| MIS\_N2N\_HO\_Query\_Resources | Command | Query available network resources | 7.4.19 |
| MIS\_MN\_HO\_Commit | Command | Notify the serving network of the decided target network information | 7.4.20 |
| MIS\_Net\_HO\_Commit | Command | Network has comitted to handover | 7.4.2 1 |
| MIS\_N2N\_HO\_Commit | Command | Notify target network that the serving network has comitted to handover | 7.4.22 |
| MIS\_MN\_HO\_Complete | Command | Initiate MN handover complete notification | 7.4.23 |
| MIS\_N2N\_HO\_Complete | Command | Handover has been completed | 7.4.24 |
| MIS\_Get\_Information | Information | Request to get information from repository | 7.4.25 |
| MIS\_Push\_Information | Information | Notify the MN of operator policies or other information | 7.4.26 |

MIS command primitives defined in MIS\_SAP indicates their destination as either the local MISF or a remote MISF. For the remote case, the local MISF will first process the primitive to create an MIS message and then forward the message to the destination peer MISF for execution. In those messages, there are TLV encoded parameters that implement the primitive parameter abstract data types within the protocol. The definition of the full binary encoding for each of these instantiations is in Annex F.

* 1. MIS\_LINK\_SAP primitives

Link\_Detected .indication

* + - * 1. Function

Link\_Detected indicates the presence of a new PoA. This implies that the MN is in the coverage area. Link\_Detected does not guarantee that the MN will be able to establish connectivity with the detected link, but just that the MN can attempt to gain connectivity. MIS users and the MISF evaluate additional properties of the link before attempting to establish an L2 connection with the link. Moreover, Link\_Detected is not generated when additional PoAs of the same link are discovered. In case of IEEE 802.11, Link\_Detected is generated by MAC state generic convergence function (MSGCF).

* + - * 1. Semantics of service primitive

Link\_Detected.indication (  
LinkDetectedInfo  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| LinkDetectedInfo | LINK\_DET\_INFO | Information of a detected link. |

* + - * 1. When generated

The Link Detected event is generated on the MN when the first PoA of an access network is detected. This event is not generated when subsequent PoAs of the same access network are discovered during the active connection on that link.

* + - * 1. Effect on receipt

The MISF receives this event from the link layer. The MISF shall pass this notification to the MIS user(s) that has subscribed for this notification. The MIS user(s), including the MISF itself, discovers additional properties of the link before selecting it for establishing connectivity.

Link\_Up.indication

* + - * 1. Function

This notification is delivered when a layer 2 connection is established on the specified link interface. All layer 2 activities in establishing the link connectivity are expected to be completed at this point of time.

* + - * 1. Semantics of service primitive

Link\_Up.indication (   
LinkIdentifier,   
OldAccessRouter,   
NewAccessRouter,   
IPRenewalFlag,   
MobilityManagementSupport  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link associated with the event. |
| OldAccessRouter | LINK\_ADDR | (Optional) Old Access Router link address. |
| NewAccessRouter | LINK\_ADDR | (Optional) New Access Router link address. |
| IPRenewalFlag | IP\_RENEWAL\_FLAG | (Optional) Indicates whether the MN needs to change IP Address in the new PoA. |
| MobilityManagementSupport | IP\_MOB\_MGMT | (Optional) Indicates the type of Mobility Manage­ment Protocol supported by the new PoA. |

* + - * 1. When generated

This notification is generated when a layer 2 connection is established for the specified link interface.

* + - * 1. Effect on receipt

The MISF shall pass this link notification to the MIS user(s) that has subscribed for this notification in an MIS\_Link\_Up event. The MIS user(s) takes different actions on this notification.

Link\_Down.indication

* + - * 1. Function

This notification is delivered when a layer 2 connection is no longer available for sending frames, that is, when the L2 connection with network is terminated and not during PoA to PoA transitions for the same network.

* + - * 1. Semantics of service primitive

Link\_Down.indication (  
LinkIdentifier,   
OldAccessRouter,   
ReasonCode  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link associated with the event. |
| OldAccessRouter | LINK\_ADDR | (Optional) Old Access Router link address. |
| ReasonCode | LINK\_DN\_REASON | Reason why the link went down. |

* + - * 1. When generated

This notification is generated when layer 2 connectivity is lost. Layer 2 connectivity is lost explicitly in  
cases where the MN initiates disassociate type procedures. In other cases, the MN can infer loss of link connectivity due to successive time-outs for acknowledgements of retransmitted packets along with loss of reception of broadcast frames.

* + - * 1. Effect on receipt

The MISF passes this link notification to the MIS user(s) that has subscribed for this notification in an MIS\_Link\_Down event. The MIS user(s) takes different actions on this notification. The handover policy function can eliminate this link from list of active links for routing connections and can consider handing over any potential active connections to other more suitable links.

Link\_Parameters\_Report.indication

* + - * 1. Function

Link\_Parameters\_Report indicates changes in link conditions that have crossed specified threshold levels. Link\_Parameters\_Report is also generated at specified intervals for various parameters.

In the case of IEEE 802.11 network, this event is generated when higher protocol layers wish to monitor the performance parameters for a network. These higher layers can be on the network side (for network initiated handovers) and MISF on the local MN can transfer these parameters. For local MN initiated handovers, the local station management entity (SME) and MSGCF would monitor link-layer properties and the MISF would normally be interested only in the Link\_Going\_Down.indication.

NOTE—The primitive to set parameter thresholds that could trigger this event is specified in 7.3.13.

* + - * 1. Semantics of service primitive

Link\_Parameters\_Report.indication(  
LinkIdentifier,   
LinkParametersReportList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link associated with the event. |
| LinkParametersReportList | LIST(LINK\_PARAM\_RPT) | A list of Link Parameter Report. |

* + - * 1. When generated

For each specified parameter, this notification is generated either at a predefined regular interval determined by a user configurable timer or when it crosses a configured threshold.

* + - * 1. Effect on receipt

The MISF receives this event from the link layer. The MISF then passes this notification to the MIS user(s) that has subscribed for this notification. The MIS user(s) takes different actions on this notification. If parameters related to link quality cross a certain threshold then that link needs to be evaluated for handing over current connections. The MISF collectively evaluates different parameters and gives appropriate indications to higher layers regarding suitability of different links.

Link\_Going\_Down.indication

* + - * 1. Function

This notification is delivered when a Layer 2 connection is expected (predicted) to go down (Link\_Down) within a certain time interval. Link\_Going\_Down event can be the indication to initiate handover procedures.

* + - * 1. Semantics of service primitive

Link\_Going\_Down.indication (   
LinkIdentifier,   
TimeInterval,   
LinkGoingDownReason  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link associated with the event. |
| TimeInterval | UNSIGNED\_INT(2) | Time Interval (in milliseconds) specifies the time interval at which the link is expected to go down. A value of ‘0’ is spec­ified if the time interval is unknown. |
| LinkGoingDownReason | LINK\_GD\_REASON | The reason why the link is going to be down. |

* + - * 1. When generated

A Link\_Going\_Down event implies that a Link\_Down is imminent within a certain time interval. If Link\_Down is NOT received within specified time interval then actions due to previous Link\_Going\_Down are ignored.

In the case of IEEE 802.11 networks, this notification is generated when the established IEEE 802.11 network connection is expected to go down within the specified time interval by the IEEE 802.11 MSGCF. The network is expected to go down because of an event whose timing is well understood, such as an explicit disconnection event observed on the MLME\_SAP. This can also be expected as the result of a predictive algorithm that monitors link quality. The details of such a predictive algorithm used are beyond the scope of this standard. This event is not generated when the IEEE 802.11 station (STA) transitions from one AP to another in the same network.

* + - * 1. Effect on receipt

The MISF receives this event from the link layer. The MISF then passes this notification to the MIS user(s) that has subscribed for this notification. MIS user(s) takes different actions on this notification. MIS users, then, prepare to initiate handovers.

Link\_Handover\_Imminent.indication (to be excluded)

Link\_Handover\_Complete.indication (to be excluded)

Link\_PDU\_Transmit\_Status.indication

* + - * 1. Function

Link\_PDU\_Transmit\_Status indicates the transmission status of a higher layer PDU by the link layer. A success status indicates that the higher layer PDU has been successfully delivered from the link layer in the local node to the link layer in the peer node. A higher layer intermediate buffer management entity could use this indication to flush the delivered PDU from its buffer. A failure status indicates that the higher layer PDU identified in the indication was not delivered successfully from the link layer in the local node to the link layer in the peer node. During a handover, if such a failure indication is received from the link connection with the source network, the higher layer intermediate buffer management entity could attempt to retransmit the failed PDU once a connection to the target network is established.

A Packet Identifier is expected to be passed alongside when each higher layer PDU is sent from the higher layer to the link for transmission. The Packet Identifier is defined in this standard as a container structure whose syntax and semantics will be decided by the upper layer (i.e., the MIS user that subscribes to this event). The MISF and link layer just pass and return the Packet Identifier and do not need to understand its syntax and semantics.

To avoid receiving excessive amount of link PDU transmission status indications, an MIS user, for example, chooses to subscribe to this event only after it receives a Link\_Handover\_Imminent.indication or when it is about to invoke an MIS\_Link\_Actions.request to perform a handover, and to unsubscribe from the event once it receives indication that the handover is completed.

* + - * 1. Semantics of service primitive

Link\_PDU\_Transmit\_Status.indication (   
LinkIdentifier,   
PacketIdentifier,   
TransmissionStatus  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link associated with the event. |
| PacketIdentifier | UNSIGNED\_INT(2) | Identifier for higher layer PDU on which this notification is generated. |
| TransmissionStatus | BOOLEAN | Status of the transmitted packet. True: Success  False: Failure |

* + - * 1. When generated

A success notification is generated when a higher layer PDU is successfully transmitted over the link. A failure notification is generated when a higher layer PDU was not transmitted successfully.

* + - * 1. Effect on receipt

The MISF receives this event from the link layer. The MISF then passes this notification to the MIS user(s) that has subscribed for this notification. The MIS user(s) takes different actions on this notification. A higher layer intermediate buffer management entity in MIS could use the success indication to flush higher layer packets stored in any intermediate buffers and a failure indication to retransmit higher layer packets stored in any intermediate buffers, especially if there are changes in the access network during handovers.

Link\_Capability\_Discover

* + - * 1. Link\_Capability\_Discover.request

Function

This primitive is used by the MISF to query and discover the list of supported link-layer events and link- layer commands.

Semantics of service primitive

No primitive parameters exist for this primitive.

Link\_Capability\_Discover.request ()

When generated

This primitive is generated by the MISF when it needs to receive link-layer event notifications and learn about which link-layer commands the lower layer can support.

Effect on receipt

The recipient responds immediately with Link\_Capability\_Discover.confirm primitive.

* + - * 1. Link\_Capabi lity\_Discover.confirm

Function

This primitive returns the result of the query to discover link-layer capability.

Semantics of service primitive

Link\_Capability\_Discover.confirm(   
Status,  
SupportedLinkEventList,   
SupportedLinkCommandList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| Status | STATUS | Status of operation. Code 3 (Authorization Failure) is not applicable. |
| SupportedLinkEventLista | LINK\_EVENT\_LIST | List of link-layer events supported by the link layer. |
| SupportedLinkCommandLista | LINK\_CMD\_LIST | List of link-layer commands supported by the link layer. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive is generated in response to a Link\_Capability\_Discover.request primitive.

Effect on receipt

The recipient examines the returned event and command list and learns about link-layer capability. However, if Status does not indicate “Success,” the recipient performs appropriate error handling.

Link\_Event\_Subscribe

* + - * 1. Link\_Event\_Subscribe.request

Function

This primitive is used by MISF (the subscriber) to subscribe an interest in one or more events from a specific link-layer technology. The response indicates which of the requested events were successfully subscribed to. Events that were not successfully subscribed to will not be delivered to the subscriber.

Semantics of service primitive

Link\_Event\_Subscribe.request (  
RequestedLinkEventList  
)

Parameter:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| RequestedLinkEventList | LINK\_EVENT\_LIST | List of link-layer events that for which the subscriber would like to receive indications. |

When generated

This primitive is generated by a subscriber such as the MISF that is seeking to receive event indications from different link-layer technologies.

Effect on receipt

The recipient responds immediately with Link\_Event\_Subscribe. confirm primitive.

* + - * 1. Link\_Event\_Subscribe.confirm

Function

This primitive returns the result of the subscription request.

Semantics of service primitive

Link\_Event\_Subscribe.confirm (  
Status,  
ResponseLinkEventList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| Status | STATUS | Status of operation. Code 3 (Authorization Failure) is not applicable. |
| ResponseLinkEventLista | LINK\_EVENT\_LIST | List of successfully subscribed link events. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive is generated in response to a Link\_Event\_Subscribe.request primitive.

Effect on receipt

The recipient examines the ResponseLinkEventList and learns about the subscription status of different events. If Status does not indicate “Success,” the recipient performs appropriate error handling.

Link\_Event\_Unsubscribe

* + - * 1. Link\_Event\_Unsubscribe.request

Function

This primitive is used by the MISF (the subscriber) to unsubscribe from a set of previously subscribed link- layer events.

Semantics of service primitive

Link\_Event\_Unsubscribe.request (   
RequestedLinkEventList  
)

Parameter:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| RequestedLinkEventList | LINK\_EVENT\_LIST | List of link-layer events for which indications need to be unsubscribed from the Event Source. |

When generated

This primitive is generated by a subscriber such as the MISF that is seeking to unsubscribe from an already subscribed set of events.

Effect on receipt

The recipient responds immediately with Link\_Event\_Unsubscribe. confirm primitive.

* + - * 1. Link\_Event\_Unsubscribe.confirm

Function

This primitive returns the result of the request to unsubscribe from receiving link-layer event notifications.

Semantics of service primitive

Link\_Event\_Unsubscribe.confirm (  
Status,  
ResponseLinkEventList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| Status | STATUS | Status of operation. Code 3 (Authorization Failure) is not applicable. |
| ResponseLinkEventLista | LINK\_EVENT\_LIST | List of successfully unsubscribed link events. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive is generated in response to a Link\_Event\_Unsubscribe.request primitive.

Effect on receipt

The recipient can examine the ResponseLinkEventList and learn about the unsubscription status of different events. If Status does not indicate “Success,” the recipient performs appropriate error handling.

Link\_Get\_Parameters

* + - * 1. Link\_Get\_Parameters.request

Function

This primitive is used by the MISF to obtain the current value of a set of link parameters of a specific link.

Semantics of service primitive

Link\_Get\_Parameters.request (  
LinkParametersRequest,  
LinkStatesRequest,  
LinkDescriptorsRequest  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| LinkParametersRequest | LIST(LINK\_PARAM\_TYPE) | A list of link parameters for which status is requested. |
| LinkStatesRequest | LINK\_STATES\_REQ | The link states to be requested. |
| LinkDescriptorsRequest | LINK\_DESC\_REQ | The link descriptors to be requested. |

When generated

This primitive is generated by the MISF to obtain the current value of a set of link parameters from a link.

Effect on receipt

The recipient link responds with Link\_Get\_Parameters.confirm primitive.

* + - * 1. Link\_Get\_Parameters.confirm

Function

This primitive is sent in response to the Link\_Get\_Parameters.request primitive. This primitive provides current value of the requested link parameters.

NOTE—How the value is measured or calculated by the link is not specified by this standard.

Semantics of service primitive

Link\_Get\_Parameters.confirm (  
Status,  
LinkParametersStatusList,  
LinkStatesResponse,  
LinkDescriptorsResponse  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| Status | STATUS | Status of operation. Code 3 (Authorization Failure) is not applicable. |
| LinkParametersStatusLista | LIST(LINK\_PARAM) | A list of measurable link parameters and their current values. |
| LinkStatesResponsea | LIST(LINK\_STATES\_RSP) | The current link state information. |
| LinkDescriptorsResponsea | LIST(LINK\_DESC\_RSP) | The descriptors of a link. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive is generated in response to the Link\_Get\_Parameters.request operation.

Effect on receipt

The recipient passes the link parameter values received to the MIS users. However, if Status does not indicate “Success,” the recipient performs appropriate error handling.

Link\_Configure\_Thresholds

* + - * 1. Link\_Configure\_Thresholds.request

Function

This primitive is used by the MISF to configure thresholds and/or specify the time interval between periodic reports for the Link\_Parameters\_Report indication.

Semantics of service primitive

Link\_Configure\_Thresholds.request (

LinkConfigureParameterList

)

Parameter:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| LinkConfigureParameterList | LIST(LINK\_CFG\_PARAM) | A list of link threshold parameters. |

When generated

This primitive is generated by an MISF that needs to set threshold values for different link parameters.

Effect on receipt

The recipient responds immediately with Link\_Configure\_Thresholds.confirm primitive.

* + - * 1. Link\_Configure\_Thresholds.confirm

Function

This primitive is sent in response to the Link\_Configure\_Thresholds.request primitive. This primitive specifies the status of threshold configuration operation.

Semantics of service primitive

Link\_Configure\_Thresholds.confirm (

Status,  
LinkConfigureStatusList   
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| Status | STATUS | Status of operation. Code 3 (Authorization Failure) is not applicable. |
| LinkConfigureStatusLista | LIST(LINK\_CFG\_STATUS) | A list of Link Configure Status. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive is generated in response to the Link\_Configure\_Thresholds.request operation.

Effect on receipt

The recipient prepares to receive Link\_Parameters\_Report indications on successful execution of this  
primitive. However, if Status does not indicate “Success,” the recipient performs appropriate error handling.

Link\_Action

* + - * 1. Link\_Action.request

Function

This primitive is used by the MISF to request an action on a link-layer connection to enable optimal handling of link-layer resources for the purpose of handovers.

The link-layer connection can be ordered (e.g., to shut down) to remain active, to perform a scan, or to come up active and remain in stand-by mode. The command execution delay time can also be specified for cases where the link-layer technology under consideration supports the action.

Semantics of service primitive

Link\_Action.request (

LinkAction,  
ExecutionDelay,  
PoALinkAddress

)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| LinkAction | LINK\_ACTION | Specifies the action to perform. |
| ExecutionDelay | UNSIGNED\_INT(2) | Time (in ms) to elapse before the action needs to be taken. A value of 0 indicates that the action is taken immediately. Time elapsed is calculated from the instance the request arrives until the time when the execution of the action is carried out. |
| PoALinkAddress | LINK\_ADDR | (Optional) The PoA link address to forward data to. This parameter is used when DATA\_FWD\_REQ action is requested. |

When generated

The MISF generates this primitive upon request from the MIS user to perform an action on a pre-defined link-layer connection.

Effect on receipt

Upon receipt of this primitive, the link-layer technology supporting the current link-layer connections performs the action specified by the Link Action parameter in accordance with the procedures specified by the relevant standards organization and at the time specified by the Execution Delay parameter.

* + - * 1. Link\_Action.confirm

Function

This primitive is used by link-layer technologies to provide an indication of the result of the action executed on the current link-layer connection.

Semantics of service primitive

Link\_Action.confirm (  
Status,  
ScanResponseSet,   
LinkActionResult   
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| Status | STATUS | Status of the operation.  Code 3 (Authorization Failure) is not applicable. |
| ScanResponseSeta | LIST(LINK\_SCAN\_RSP) | (Optional) A list of discovered links and related information. |
| LinkActionResulta | LINK\_AC\_RESULT | Specifies whether the link action was successful. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

The link-layer technology generates this primitive to communicate the result of the action executed on the link-layer connection.

Effect on receipt

Upon receipt of this primitive, the MISF determines the relevant MIS command that needs to be used to provide an indication or confirmation to the MIS user of the actions performed on the current link-layer connection. If a Scan action was issued by the associated Link\_Action.request, the optional ScanResponseSet field is included in the Link\_Action.confirm response.

* 1. MIS\_SAP primitives

The primitives defined as part of MIS\_SAP are described in the following subclauses.

MIS\_Capability\_Discover

* + - * 1. MIS\_Capability\_Discover.request

Function

This primitive is used by an MIS user to discover the capabilities of the local MISF or a remote MISF. When invoking this primitive to discover the capabilities of a remote MISF, the MIS user can optionally piggyback the capability information of its local MISF so that the two MISFs can mutually discover each other’s capabilities with a single invocation of this primitive.

Semantics of service primitive

MIS\_Capability\_Discover.request (  
DestinationIdentifier,  
LinkAddressList,  
SupportedMisEventList,  
SupportedMisCommandList,  
SupportedIsQueryTypeList,  
SupportedTransportList,  
MBBHandoverSupport,

SupportedSecurityCapList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies the local MISF or a remote MISF that will be the destination of this request. |
| LinkAddressList | LIST(NET\_TYPE\_ADDR) | (Optional) A list of network type and link address pair on the local MISF. |
| SupportedMISEventList | MIS\_EVT\_LIST | (Optional) List of supported events on the local MISF. |
| SupportedMISCommandList | MIS\_CMD\_LIST | (Optional) List of supported commands on the local MISF. |
| SupportedISQueryTypeList | MIS\_IQ\_TYPE\_LST | (Optional) List of supported MIIS query types on the local MISF. |
| SupportedTransportList | MIS\_TRANS\_LST | (Optional) List of supported transport types on the local MISF. |
| MBBHandoverSupport | LIST(MBB\_HO\_SUPP) | (Optional) This is used to indicate if a make before break handover is supported on the local MISF. Break before make handover is always supported. |
| SupportedSecurityCapList | MIS\_SEC\_CAP | (Optional) List of supported MIS security capabilities on the local MISF. |

When generated

This primitive is generated by an MIS user to discover the capabilities of the local MISF or a remote MISF. In the case of remote discovery, this primitive contains the SupportedMisEventList, SupportedMisCommandList, SupportedIsQueryTypeList, SupportedTransportList, and MBBHandoverSupport parameters of the local MISF to enable mutual discovery of each other’s capabilities.

Effect on receipt

If the destination of the request is the local MISF itself, the local MISF responds with MIS\_Capability\_Discover.confirm. If the destination of the request is a remote MISF, the local MISF shall generate a corresponding MIS\_Capability\_Discover request message to the remote MISF if it does not have the capability information of the remote MISF.

* + - * 1. MI H\_Capability\_Discover. indication

Function

This primitive is used by an MISF to notify an MIS user on the receipt of an MIS\_Capability\_Discover request message from a peer MISF.

Semantics of service primitive

MIS\_Capability\_Discover.indication (

SourceIdentifier  
LinkAddressList,  
SupportedMisEventList,  
 SupportedMisCommandList,   
SupportedIsQueryTypeList,  
SupportedTransportList,  
MBBHandoverSupport,

SupportedSecurityCapList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which is a remote MISF. |
| LinkAddressList | LIST (NET\_TYPE\_ADDR) | (Optional) A list of network type and link address pair on the remote MISF. |
| SupportedMISEventList | MIS\_EVT\_LIST | (Optional) List of supported events on the remote MISF. |
| SupportedMISCommandList | MIS\_CMD\_LIST | (Optional) List of supported commands on the remote MISF. |
| SupportedISQueryTypeList | MIS\_IQ\_TYPE\_LST | (Optional) List of supported MIIS query types on the remote MISF. |
| SupportedTransportList | MIS\_TRANS\_LST | (Optional) List of supported transport types on the remote MISF. |
| MBBHandoverSupport | LIST (MBB\_HO\_SUPP) | (Optional) This is used to indicate if a make before break handover is supported on the remote MISF. Break before make handover is always supported. |
| SupportedSecurityCapList | MIS\_SEC\_CAP | (Optional) List of supported MIS security capabilities on the remote MISF. |

When generated

This primitive is used by an MISF to notify an MIS user when an MIS\_Capability\_Discover request message is received. This primitive is optional since the MISF can immediately return an MIS\_Capability\_Discover response message without generating this primitive to the MIS user.

Effect on receipt

The MIS user responds with an MIS\_Capability\_Discover.response primitive when an indication is received.

* + - * 1. MIH\_Capability\_Discover.response

Function

This primitive is used by an MIS user to convey the locally supported MIS capabilities to the MIS user that invoked the MIS\_Capability\_Discover request.

Semantics of Service primitive

MIS\_Capability\_Discover.response(  
DestinationIdentifier,  
Status,  
LinkAddressList,  
SupportedMisEventList,  
SupportedMisCommandList,  
SupportedIsQueryTypeList,  
SupportedTransportList,  
MBBHandoverSupport,

SupportedSecurityCapList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies the remote MISF that will be the destination of this response. |
| Status | STATUS | Status of operation. |
| LinkAddressList | LIST (NET\_TYPE\_ADDR) | (Optional) A list of network type and link address pair on local MISF. |
| SupportedMISEventList | MIS\_EVT\_LIST | (Optional) List of supported events on local MISF. |
| SupportedMISCommandList | MIS\_CMD\_LIST | (Optional) List of supported commands on local MISF. |
| SupportedISQueryTypeList | MIS\_IQ\_TYPE\_LST | (Optional) List of supported MIIS query types on local MISF. |
| SupportedTransportList | MIS\_TRANS\_LST | (Optional) List of supported transport types on local MISF. |
| MBBHandoverSupport | LIST (MBB\_HO\_SUPP) | (Optional) This is used to indicate if a make before break handover is supported on local MISF. Break before make handover is always supported. |
| SupportedSecurityCapList | MIS\_SEC\_CAP | (Optional) List of supported MIS security capabilities on the local MISF. |

When generated

This primitive is generated by an MIS user as a response to a received MIS\_Capability\_Discover.indication primitive.

Effect on receipt

Upon receiving this primitive, the MISF shall generate and send the corresponding MIS\_Capability\_Discover response message to the destination MISF.

* + - * 1. MIS\_Capability\_Discover.confirm

Function

This primitive is used by the MISF to convey the supported MIS capabilities about Event Service, Command Service, and Information Service to the MIS user that invoked the MIS\_Capability\_Discover.request.

Semantics of service primitive

MIS\_Capability\_Discover.confirm (  
SourceIdentifier,  
Status,  
LinkAddressList,  
SupportedMisEventList,  
SupportedMisCommandList,  
SupportedIsQueryTypeList,  
SupportedTransportList,  
MBBHandoverSupport,

SupportedSecurityCapList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| Status | STATUS | Status of operation. |
| LinkAddressList | LIST (NET\_TYPE\_ADDR) | (Optional) A list of network type and link address pair on the MISF identified by Source Identifier. |
| SupportedMISEventList | MIS\_EVT\_LIST | (Optional) List of supported events on the MISF identified by Source Identifier. |
| SupportedMISCommandList | MIS\_CMD\_LIST | (Optional) List of supported commands on the MISF identified by Source Identifier. |
| SupportedISQueryTypeList | MIS\_IQ\_TYPE\_LST | (Optional) List of supported MIIS query types on the MISF identified by Source Identifier. |
| SupportedTransportList | MIS\_TRANS\_LST | (Optional) List of supported transport types on the MISF identified by Source Identifier. |
| MBBHandoverSupport | LIST (MBB\_HO\_SUPP) | (Optional) This is used to indicate if a make before break handover is supported on the MISF identified by Source Identifier. Break before make handover is always supported. |
| SupportedSecurityCapList | MIS\_SEC\_CAP | (Optional) List of supported MIS security capabilities on the remote MISF. |

When generated

This primitive is invoked by a local MISF to convey the results of a previous MIS\_Capability\_Discover.request primitive from an MIS user.

Effect on receipt

Upon reception of this primitive the receiving entity becomes aware of the supported MIS capabilities. However, if Status does not indicate “Success,” the recipient ignores any other returned values and, instead, performs appropriate error handling.

MIS\_Register

* + - * 1. MIS\_Register.request

Function

This primitive is used by an MIS user to register the local MISF with remote MISF.

Semantics of service primitive

MIS\_Register.request (  
DestinationIdentifier,  
LinkIdentifierList,  
RequestCode  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies a remote MISF that will be the destination of this request. |
| LinkIdentifierList | LIST(LINK\_ID) | List of local link identifiers. |
| RequestCode | REG\_REQUEST\_CODE | Registration request code. Depending on the request code, the MIS user can choose to either register or re-register with the remote MISF. |

When generated

This primitive is invoked by the MIS user when it needs to register the local MISF with a remote MISF.

Effect on receipt

On receipt, the local MISF sends an MIS\_Register request message to the destination MISF.

* + - * 1. MIS\_Register.indication

Function

This primitive is used by an MISF to notify an MIS user that an MIS\_Register request message has been received.

Semantics o f service primitive

MIS\_Register.indication (  
SourceIdentifier,  
LinkIdentifierList,  
RequestCode  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which is a remote MISF. |
| LinkIdentifierList | LIST(LINK\_ID) | List of link identifiers of the remote MISF. |
| RequestCode | REG\_REQUEST\_CODE | Registration request code. Depending on the request code, the MIS user can choose to either register or re-register with the remote MISF. |

When generated

This primitive is generated by the remote MISF when an MIS\_Register request message is received.

Effect on receipt

The remote MIS user will perform necessary actions to process the registration request and respond with an MIS\_Register.response.

* + - * 1. MIS\_Register.response

Function

This primitive is used by an MIS user to send the processing status of a received registration request.

Semantics of service primitive

MIS\_Register.response (  
DestinationIdentifier,  
Status,  
ValidTimeInterval  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies a remote MISF, which will be the destination of this response. |
| Status | STATUS | Status of operation. |
| ValidTimeIntervala | UNSIGNED\_INT(4) | Time interval in seconds during which the registration is valid. Parameter applicable only when the status parameter indicates a successful operation. A value of 0 indicates an infinite validity period. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive is invoked by the MIS user to report back the result after completing the processing of a registration request.

Effect on receipt

Upon receipt, the local MISF sends an MIS\_Register response message to the destination MISF.

* + - * 1. MIS\_Register.confirm

Function

This primitive is used by the local MISF to convey the result of a registration request to an MIS user.

Semantics of service primitive

MIS\_Register.confirm (

SourceIdentifier,  
Status,  
ValidTimeInterval  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which is a remote MISF. |
| Status | STATUS | Status of operation. |
| ValidTimeIntervala | UNSIGNED\_INT(4) | Time interval in seconds during which the registration is valid. Parameter applicable only when the status parameter indicates a successful operation. A value of 0 indicates an infinite validity period. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive is used by an MISF to notify an MIS user the result of an MIS registration request.

Effect on receipt

Upon receipt, the MIS user can determine the result of the registration request.

MIS\_DeRegister

* + - * 1. MIS\_DeRegister.request

Function

This primitive is used by an MIS user to deregister the local MISF with peer MISF.

Semantics of service primitive

MIS\_DeRegister.request (  
DestinationIdentifier  
)

Parameter:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies a remote MISF that will be the destination of this request. |

When generated

This primitive is invoked by the MIS user when it needs to terminate an existing MIS registration with a remote MISF.

Effect on receipt

Upon receipt, the local MISF generates and sends an MIS\_DeRegister request message to the destination MISF.

* + - * 1. MIS\_DeRegister.indication

Function

This primitive is used by an MISF to notify an MIS user that an MIS\_DeRegister request message has been received.

Semantics of service primitive

MIS\_DeRegister.indication(  
SourceIdentifier )  
)

Parameter:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which is a remote MISF. |

When generated

This primitive is generated by an MISF when an MIS\_DeRegister request message is received.

Effect on receipt

The MIS user will perform necessary actions to process the deregistration request and respond with an MIS\_DeRegister.response.

* + - * 1. MIS\_DeRegister.response

Function

This primitive is invoked by a remote MIS user to respond with the processing status of a received deregistration request.

Semantics of service primitive

MIS\_DeRegister.response (  
DestinationIdentifier,  
Status  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies a remote MISF, which will be the destination of this response. |
| Status | STATUS | Status of operation. Code 2 (Reject) is not used. |

When generated

This primitive is invoked by the MIS user to report back the result after completing the processing of a deregistration request from a remote MIS user.

Effect on receipt

Upon receipt, the local MISF sends an MIS\_DeRegister response message to the destination MISF.

* + - * 1. MIS\_DeRegister.confirm

Function

This primitive is used by the local MISF to convey the result of a deregistration request to the local MIS user.

Semantics of service primitive

MIS\_DeRegister.confirm (  
SourceIdentifier,  
Status  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which is a remote MISF. |
| Status | STATUS | Status of operation. Code 2 (Rejected) is not used. |

When generated

This primitive is used by an MISF to notify the local MIS user the status of MIS deregistration request.

Effect on receipt

Upon receipt, the MIS user can determine the status of the deregistration request.

MIS\_Event\_Subscribe

* + - * 1. MIS\_Event\_Subscribe.request

Function

This primitive is used by an MIS user (the subscriber) to subscribe an interest in one or more MIS event types from the local or a remote MISF. Optionally, the subscriber indicates a list of specific configuration information applicable for various events being subscribed. If configured, the event must be triggered only when all the criteria set in the parameters are met.

Semantics of service primitive

MIS\_Event\_Subscribe.request (  
DestinationIdentifier,  
LinkIdentifier,  
RequestedMisEventList,  
EventConfigurationInfoList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies the local MISF or a remote MISF that will be the destination of this request. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link for event subscription. For local event subscription, PoA link address need not be present if the link type lacks such a value. |
| RequestedMISEventList | MIS\_EVT\_LIST | List of MIS events that the endpoint would like to receive indications for, from the Event Source. |
| EventConfigurationInfoList | LIST(EVT\_CFG\_INFO) | (Optional) List of additional configuration informa­tion for event subscription. |

When generated

This primitive is invoked by an MIS user when it wants to receive indications on a set of specific MIS events from the local MISF or a remote MISF.

Effect on receipt

If the destination of the request is the local MISF itself, the local MISF responds immediately with an MIS\_Event\_Subscribe.confirm primitive. If the destination of the request is a remote MISF, the local MISF generates and sends an MIS\_Event\_Subscribe request message to the remote MISF.

* + - * 1. MIS\_Event\_Subscribe.confirm

Function

This primitive returns the result of an MIS event subscription request.

Semantics of service primitive

MIS\_Event\_Subscribe.confirm (  
SourceIdentifier,  
Status,  
LinkIdentifier,  
ResponseMisEventList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| Status | STATUS | Status of operation. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link for event subscription. |
| ResponseMISEventLista | MIS\_EVT\_LIST | List of successfully subscribed MIS events. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive is generated by the local MISF at the completion of processing an MIS\_Event\_Subscribe.request primitive from a local MIS user or in response to the receiving of an MIS\_Event\_Subscribe response message from a peer MISF.

Effect on receipt

The recipient MIS user examines the returned event list and learns about the subscription status of different events. However, if Status does not indicate “Success,” the recipient performs appropriate error handling.

MIS\_Event\_Unsubscribe

* + - * 1. MIS\_Event\_Unsubscribe.request

Function

This primitive is used by an MIS user (the subscriber) to unsubscribe from a set of previous subscribed MIS events.

Semantics of service primitive

MIS\_Event\_Unsubscribe.request (  
DestinationIdentifier,  
LinkIdentifier,  
RequestedMisEventList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies the local MISF or a remote MISF, which will be the destination of this request. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link for event unsubscription. For local event unsubscription, PoA address in the Link Identifier need not be present if the link type lacks such a value. |
| RequestedMISEventList | MIS\_EVT\_LIST | List of MIS events for which indications need to be unsub­scribed from the Event Source. |

When generated

This primitive is invoked by an MIS user (subscriber) that is seeking to unsubscribe from an already subscribed set of events from the local MISF or a remote MISF.

Effect on receipt

If the destination of the request is the local MISF itself, the local MISF responds immediately with MIS\_Event\_Unsubscribe.confirm primitive. If the destination of the request is a remote MISF, the local MISF generates and sends an MIS\_Event\_Unsubscribe request message to the remote MISF.

* + - * 1. MIS\_Event\_Unsubscribe.confirm

Function

This primitive returns the result of an MIS event unsubscription request.

Semantics of service primitive

MIS\_Event\_Unsubscribe.confirm (  
SourceIdentifier,  
Status,  
LinkIdentifier,  
ResponseMisEventList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| Status | STATUS | Status of operation. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link for event unsubscription. |
| ResponseMISEventLista | MIS\_EVT\_LIST | List of successfully unsubscribed link events. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive is generated by the local MISF at the completion of processing an MIS\_Event\_Unsubscribe.request primitive from a local MIS user or in response to the receiving of an MIS\_Event\_Unsubscribe response message from a peer MISF.

Effect on receipt

The recipient MIS user can examine the returned event list and learn about the unsubscription status of different events. However, if Status does not indicate “Success,” the recipient performs appropriate error handling.

MIS\_Link\_Detected.indication

* + - * 1. Function

The MIS\_Link\_Detected.indication is sent to local MISF users to notify them of a local event or of a receipt of MIS\_Link\_Detected indication message from a remote MISF.

* + - * 1. Semantics of the service primitive

MIS\_Link\_Detected.indication (

SourceIdentifier,  
LinkDetectedInfoList   
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| LinkDetectedInfoList | LIST(LINK\_DET\_INFO) | List of link detection information. |

* + - * 1. When generated

The MIS\_Link\_Detected.indication is sent to local MISF users to notify them of a local event (i.e., Link\_Detected.indication), or of receipt of MIS\_Link\_Detected indication message from a remote MISF (i.e., a remote Link\_Detected event has occurred).

* + - * 1. Effect on receipt

MIS user dependant.

MIS\_Link\_Up.indication

* + - * 1. Function

The MIS\_Link\_Up.indication is sent to local MISF users to notify them of a local event, or is the result of the receipt of an MIS\_Link\_Up indication message to indicate to the remote MISF users who have subscribed to this remote event.

* + - * 1. Semantics of the service primitive

MIS\_Link\_Up.indication (  
SourceIdentifier,  
LinkIdentifier,  
OldAccessRouter,  
NewAccessRouter,  
IPRenewalFlag,  
Mobility Management Support  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link associated with the event. |
| OldAccessRouter | LINK\_ADDR | (Optional) Link address of old Access Router. |
| NewAccessRouter | LINK\_ADDR | (Optional) Link address of new Access Router. |
| IPRenewalFlag | IP\_RENEWAL\_FLAG | (Optional) Indicates whether the MN needs to change IP Address in the new PoA. |
| MobilityManagementSupport | IP\_MOB\_MGMT | (Optional) Indicates the type of Mobility Management Protocol supported by the new PoA. |

* + - * 1. When generated

The MIS\_Link\_Up.indication is sent to local MISF users to notify them of a local event (i.e., Link\_Up.indication), or is the result of the receipt of an MIS\_Link\_Up indication message to indicate to the remote MISF users who have subscribed to this remote event that a remote link up event occurred.

* + - * 1. Effect on receipt

MIS user dependant.

MI H\_Link\_Down .ind ication

* + - * 1. Function

The MIS\_Link\_Down.indication is sent to local MISF users to notify them of a local event, or is the result of the receipt of an MIS\_Link\_Down indication message to indicate to the remote MISF users who have subscribed to this remote event.

* + - * 1. Semantics of the service primitive

MIS\_Link\_Down.indication (

SourceIdentifier,  
LinkIdentifier,  
OldAccessRouter,  
ReasonCode  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link associated with the event. |
| OldAccessRouter | LINK\_ADDR | (Optional) Link address of old Access Router. |
| ReasonCode | LINK\_DN\_REASON | Reason why the link went down. |

* + - * 1. When generated

The MIS\_Link\_Down.indication is sent to local MISF users to notify them of a local event (i.e.,  
Link\_Down.indication), or is the result of the receipt of an MIS\_Link\_Down indication message to indicate to the remote MISF users who have subscribed to this remote event that a remote link\_down event occurred.

* + - * 1. 4 Effect on receipt

MIS user dependant.

MIS\_Link\_Parameters\_Report. indication

* + - * 1. Function

MIS\_Link\_Parameters\_Report indication is sent by the local MISF to a local MIS user to report the status of a set of parameters of a local or remote link. This MIS event is either local or remote.

* + - * 1. Semantics of service primitive

MIS\_Link\_Parameters\_Report.Indication (  
SourceIdentifier,  
LinkIdentifier,  
LinkParameterReportList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link associated with the event. |
| LinkParameterReportList | LIST(LINK\_PARAM\_RPT) | A list of Link Parameter Reports. |

* + - * 1. When generated

This notification is generated by the local MISF either

* At a predefined regular interval determined by a user configurable timer;
* When a specified parameter of a currently active local interface crosses a configured threshold. In such a case, the local MISF most likely will first receive a Link\_Parameters\_Report. indication from the local link layer; or
* When an MIS\_Link\_Parameters\_Report indication message is received from a remote MISF.
  + - * 1. Effect on receipt

Upper layer entities take different actions upon receipt of this indication.

MIS\_Link\_Going\_Down.indication

* + - * 1. Function

The MIS\_Link\_Going\_Down.indication is sent to local MISF users to notify them of a local event, or is the result of the receipt of an MIS\_Link\_Going\_Down indication message to indicate to the remote MISF users who have subscribed to this remote event.

* + - * 1. Semantics of the service primitive

MIS\_Link\_Going\_Down.indication (

SourceIdentifier,  
LinkIdentifier,  
TimeInterval,  
LinkGoingDownReason  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link associated with the event. |
| TimeInterval | UNSIGNED\_INT(2) | Time Interval (in milliseconds) specifies the time interval at which the link is expected to go down. A value of ‘0’ is specified if time interval is unknown or uncertain. |
| LinkGoingDownReason | LINK\_GD\_REASON | The reason why the link is going down. |

* + - * 1. When generated

The MIS\_Link\_Going\_Down.indication is sent to local MISF users to notify them of a local event (i.e., Link\_Going\_Down.indication), or is the result of the receipt of an MIS\_Link\_Going\_Down indication message to indicate to the remote MISF users who have subscribed to this remote event that a remote link\_going\_down event occurred.

* + - * 1. Effect on receipt

MIS user dependant.

MIS\_Link\_Handover\_Imminent.indication

* + - * 1. Function

This primitive is issued by the MISF to report the imminent occurrence of an intra-technology link handover. This MIS event is either local or remote. This indication directly corresponds to the link-layer event Link\_Handover\_Imminent.indication defined in 7.3.6.

* + - * 1. Semantics of service primitive

MIS\_Link\_Handover\_Imminent.indication (  
SourceIdentifier,  
OldLinkIdentifier,  
NewLinkIdentifier,  
OldAccessRouter,  
NewAccessRouter  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| OldLinkIdentifier | LINK\_TUPLE\_ID | Identifier of the old link. |
| NewLinkIdentifier | LINK\_TUPLE\_ID | Identifier of the new link. |
| OldAccessRouter | LINK\_ADDR | (Optional) Link address of old Access Router. |
| NewAccessRouter | LINK\_ADDR | (Optional) Link address of new Access Router. |

* + - * 1. When generated

This notification is generated by the MISF when a link-layer intra-technology handover is about to occur. The event could be triggered by the reception of a Link\_Handover\_Imminent.indication from a link or on receipt of an MIS\_Link\_Handover\_Imminent indication message.

* + - * 1. Effect on receipt

Upper layer entities take different actions upon notification.

MIH\_Link\_Handover\_Complete.indication

* + - * 1. Function

This primitive is issued by the MISF to report the completion of an intra-technology link handover. This MIS event is either local or remote. MIS\_Link\_Handover\_Complete indication is a result of a Link\_Handover\_Complete indication from the link layer.

* + - * 1. Semantics of service primitive

MIS\_Link\_Handover\_Complete.indication (  
SourceIdentifier,  
OldLinkIdentifier,  
NewLinkIdentifier,  
OldAccessRouter,  
NewAccessRouter,  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| OldLinkIdentifier | LINK\_TUPLE\_ID | Identifier of the old link. |
| NewLinkIdentifier | LINK\_TUPLE\_ID | Identifier of the new link. |
| OldAccessRouter | LINK\_ADDR | (Optional) Link address of old Access Router. |
| NewAccessRouter | LINK\_ADDR | (Optional) Link address of new Access Router. |

* + - * 1. When generated

This notification is generated by the MISF when a link-layer intra-technology handover is completed. The event could be triggered by the reception of a Link\_Handover\_Complete. indication from a link or on receipt of an MIS\_Link\_Handover\_Complete indication message.

* + - * 1. Effect on receipt

Upper layer entities take different actions on this notification. An MIS user makes use of this notification to configure other layers (IP, Mobile IP) for various upper layer handovers that are needed. Transport layers (e.g., TCP) also make use of this primitive to fine tune their flow control and flow congestion mechanisms.

MIS\_Link\_PD U\_Transmit\_Status.indication

* + - * 1. Function

The MIS\_Link\_PDU\_Transmit\_Status.indication is sent to local MISF users to notify them of a local event.

* + - * 1. Semantics of the service primitive

MIS\_Link\_PDU\_Transmit\_Status.indication(

SourceIdentifier,  
LinkIdentifier,  
PacketIdentifier,  
TransmissionStatus  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the local MISF where this event occurred. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link associated with the event. |
| PacketIdentifier | UNSIGNED\_INT(2) | Identifier for higher layer PDU on which this notification is generated. |
| TransmissionStatus | BOOLEAN | Status of the transmitted packet:  True: Success False: Failure |

* + - * 2. When generated

The MIS\_Link\_PDU\_Transmit\_Status.indication is sent to local MISF users to notify them of a local event (i.e., Link\_PDU\_Transmit\_Status.indication).

* + - * 1. Effect on receipt

MIS user dependent.

MIS\_Link\_Get\_Parameters

* + - * 1. General

A MIS\_Link\_Get\_Parameters command is issued by upper layer entities to discover and monitor the status of the currently connected and potentially available links. This command is also used to get device state information. The destination of a MIS\_Link\_Get\_Parameters command is local or remote. For example, an MIS\_Link\_Get\_Parameters request issued by a local upper layer helps the policy function that resides out of the MIS to make optimal handover decisions for different applications when multiple links are available in an MN. However, a remotely initiated MIS\_Link\_Get\_Parameters request from the network side enables the network to collect the status information on multiple links in an MN through the currently connected link.

* + - * 1. MIS\_Link\_Get\_Parameters.request

Function

This primitive is invoked by an MIS user to discover the status of the currently connected and potentially available links.

Semantics of the service primitive

MIS\_Link\_Get\_Parameters.request (

DestinationIdentifier,  
DeviceStatesRequest,  
LinkIdentifierList,  
GetStatusRequestSet  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies the local MISF or a remote MISF that will be the destination of this request. |
| DeviceStatesRequest | DEV\_STATES\_REQ | (Optional) List of device states being requested. |
| LinkIdentifierList | LIST(LINK\_ID) | List of link identifiers for which status is requested. If the list is empty, return the status of all available links. |
| GetStatusRequestSet | LINK\_STATUS\_REQ | Indicate which link status(es) is being requested. |

When generated

This primitive is invoked by an MIS user when it wants to request the status information of a set of local or remote links.

Effect of receipt

If the destination of the request is the local MISF itself, the local MISF gets the requested information on the status of the specified local links and responds with an MIS\_Link\_Get\_Parameters.confirm. If the destination of the request is a remote MISF, the local MISF generates and sends an MIS\_Link\_Get\_Parameters request message to the remote MISF.

* + - * 1. MIS\_Link\_Get\_Parameters.confirm

Function

This primitive is issued by an MISF to report the requested status of a set of specific local or remote links in response to an MIS\_Link\_Get\_Parameters request from a local or remote MIS user.

Semantics of the service primitive

MIS\_Link\_Get\_Parameters.confirm (

SourceIdentifier,  
Status,  
DeviceStatesResponseList,  
GetStatusResponseList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| Status | STATUS | Status of operation. |
| DeviceStatesResponseLista | LIST (DEV\_STATES\_RSP) | (Optional) List of device states responses. |
| GetStatusResponseLista | LIST (  SEQUENCE(LINK\_ID, LINK\_STATUS\_RSP)  ) | List of link status responses. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive returns the results of an MIS\_Link\_Get\_Parameters request to the requesting MIS user.

Effect of receipt

Upon receipt of the link status information, the MIS user makes appropriate decisions and takes suitable actions. However, if Status does not indicate “Success,” the recipient performs appropriate error handling.

MIS\_Link\_Configure\_Thresholds

* + - * 1. General

The MIS\_Link\_Configure\_Thresholds is issued by an upper layer entity to configure parameter report thresholds of a lower layer. The destination of an MIS\_Link\_Configure\_Thresholds command is local or remote. This command configures one or more thresholds on a link. When a given threshold is crossed, an MIS\_Link\_Parameters\_Report notification shall be sent to all MIS users that are subscribed to this threshold-crossing event.

* + - * 1. MIS\_Link\_Configure\_Thresholds.request

Function

This primitive is issued by an MIS user to configure thresholds of a lower layer link.

Semantics of the service primitive

MIS\_Link\_Configure\_Thresholds.request (

DestinationIdentifier,  
LinkIdentifier,  
ConfigureRequestList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies the local MISF or a remote MISF that will be the destination of this request. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link to be configured. |
| ConfigureRequestList | LIST (LINK\_CFG\_PARAM) | A list of link threshold parameters. |

When generated

This primitive is invoked by an MIS user when it attempts to configure thresholds of a local or remote lower layer link.

Effect of receipt

If the destination of the request is the local MISF itself, the local MISF issues a Link\_Configure\_Thresholds request to the lower layer link to set the thresholds for the link according to the specified configuration parameters.

If the destination of the request is a remote MISF, the local MISF generates and sends an MIS\_Link\_Configure\_Thresholds request message to the remote MISF. Upon the receipt of the message, the remote MISF then issues a Link\_Configure\_Thresholds request to the lower layer link to set the thresholds for the link according to the specified configuration parameters.

* + - * 1. MIS\_Link\_Configure\_Thresholds.confirm

Function

This primitive is issued by an MISF to report the result of an MIS\_Link\_Configure\_Thresholds request.

Semantics of the service primitive

MIS\_Link\_Configure\_Thresholds.confirm (

SourceIdentifier,  
Status,  
LinkIdentifier,  
ConfigureResponseList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| Status | STATUS | Status of operation. |
| LinkIdentifier | LINK\_TUPLE\_ID | Identifier of the link configured. |
| ConfigureResponseLista | LIST (LINK\_CFG\_STATUS) | A list of the configuration status for each requested link threshold parameter. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive returns the result of an MIS\_Link\_Configure\_Thresholds request to the requesting MIS user.

Effect of receipt

Upon receipt of the result, the MIS user makes appropriate evaluations and takes any suitable actions. However, if Status does not indicate “Success,” the recipient performs appropriate error handling.

MIS\_Link\_Actions

* + - * 1. MIS\_Link\_Actions.request

Function

This primitive is used by an MIS user to control the behavior of a set of local or remote lower layer links.

Semantics of service primitive

The parameters of the service primitive are as follows:

MIS\_Link\_Actions.request (

Destination Identifier,  
LinkActionsList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies the local MISF or a remote MISF that will be the destination of this request. |
| LinkActionsList | LIST (LINK\_ACTION\_REQ) | Specifies the suggested actions. |

When generated

This primitive is invoked by an MIS user when it attempts to control the behavior of a set of local or remote lower layer links.

Effect on receipt

If the destination of the request is the local MISF itself, the local MISF issues Link\_Action.request(s) to the specified lower layer link(s).

If the destination of the request is a remote MISF, the local MISF generates and sends an MIS\_Link\_Actions request message to the remote MISF. Upon the receipt of the message, the remote MISF then issues Link\_Action.request(s) to the specified lower layer link(s).

* + - * 1. MIS\_Link\_Actions.confirm

Function

This primitive is issued by an MISF to report the result of an MIS\_Link\_Actions request.

Semantics of the service primitive [CEP: STOPPED HERE]

The parameters of the primitive are as follows:

MIS\_Link\_Actions.confirm (

SourceIdentifier,  
Status,  
LinkActionsResultList  
 )

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker of this primitive, which can be either the local MISF or a remote MISF. |
| Status | STATUS | Status of operation. |
| LinkActionsResultLista | LIST(LINK\_ACTION\_RSP) | Contain the result of the request link actions. |

aThis parameter is not included if Status does not indicate “Success.”

When generated

This primitive returns the result of an MIS\_Link\_Actions.request to the requesting MIS user.

Effect on receipt

Upon receipt of the result, the MIS user makes appropriate evaluations and takes any suitable actions. However, if Status does not indicate “Success,” the recipient performs appropriate error handling.

MIS\_Net\_HO\_Candidate\_Query (to be excluded)

MIS\_MN\_HO\_Candidate\_Query (to be excluded)

MIS\_N2N\_HO\_Query\_Resources (to be excluded)

MIS\_MN\_HO\_Commit (to be excluded)

MIS\_Net\_HO\_Commit (to be excluded)

MIS\_N2N\_HO\_Commit (to be excluded)

MIS\_MN\_HO\_Complete (to be excluded)

MIS\_N2N\_HO\_Complete (to be excluded)

MIS\_Get\_Information

* + - * 1. MIS\_Get\_Information.request

Function

This primitive is used by an MIS user to request information from an MIS information server. The information query is related to a specific interface, attributes to the network interface, as well as the entire network capability. The service primitive has the flexibility to query either a specific data within a network interface or extended schema of a given network. It is assumed that the available information could be broadcast in access technology specific manner such as in IEEE Std 802.11 and IEEE Std 802.16.

Semantics of service primitive

MIS\_Get\_Information.request (

DestinationIdentifier,  
InfoQueryBinaryDataList,  
InfoQueryRDFDataList,  
InfoQueryRDFSchemaURL,  
InfoQueryRDFSchemaList,  
MaxResponseSize,  
QuerierNetworkType,  
UnauthenticatedInformationRequest  
)

Parameters:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | | **Data type** | | **Description** | |
| Destination Identifier | | MISF\_ID | | The local MISF or a remote MISF that will be the destination of this request. | |
| InfoQueryBinaryDataList | | LIST(IQ\_BIN\_DATA) | | (Optional) A list of TLV queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be pro­cessed by MIIS. See Table F.15 for detailed definition. | |
| InfoQueryRDFDataList | | LIST(IQ\_RDF\_DATA) | | (Optional) A list of RDF queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be pro­cessed by MIIS. See Table F.16 for detailed definition. | |
| InfoQueryRDFSchemaURL | | BOOLEAN | | (Optional) A RDF Schema URL query. This field is required only when the value is “TRUE,” which indi­cates to query a list of RDF schema URLs. | |
| InfoQueryRDFSchemaList | | LIST(IQ\_RDF\_SCHM) | | (Optional) A list of RDF schema queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be processed by MIIS. | |
| MaxResponseSize | | UNSIGNED\_INT(2) | | (Optional) This field specifies the maximum size of Info Response parameters in MIS\_Get\_Information response primitive in octets. If this field is not speci­fied, the maximum size is set to 65 535. The actual maximum size forced by the IS server can be smaller than that specified by the IS client. | |
| QuerierNetworkType | | NETWORK\_TYPE | | (Optional) The type of the network being used by the querier. This parameter is valid only with InfoQuery­BinaryDataList and InfoQueryRDFDataList. | |
| UnauthenticatedInformation- Request | | BOOLEAN | | The value of UIR bit to be set in the MIS\_Get\_Information request message sent to the remote MISF. | |

One and only one of the following parameters is specified:

* InfoQueryBinaryDataList
* InfoQueryRDFDataList
* InfoQueryRDFSchemaURL
* InfoQueryRDFSchemaList

When generated

This primitive is generated by an MIS user that is seeking to retrieve information.

The order of the queries in each of InfoQueryBinaryDataList, InfoQueryRDFDataList, and InfoQueryRDF SchemaList parameters identifies the priority of the query. The first query has the highest priority to be processed by MIIS.

Effect on receipt

If the DestinationIdentifer contains a remote MISF, then the recipient shall forward the query in an MIS\_Get\_Information request message to the designated MIIS server. If the DestinationIdentifer is for the local MISF, then the recipient shall interpret the query request and retrieve the specified information.

* + - * 1. MIS\_Get\_Information. indication

Function

This primitive is used by the MISF to indicate that an MIS\_Get\_Information request message is received from a peer MISF.

Semantics of service primitive

MIS\_Get\_Information.indication (

SourceIdentifier,  
InfoQueryBinaryDataList,  
InfoQueryRDFDataList,  
InfoQueryRDFSchemaURL,  
InfoQueryRDFSchemaList,  
MaxResponseSize,  
QuerierNetworkType,  
UnauthenticatedInformationRequest  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | Specifies the MISF ID of the node that sent the MIS\_GET\_Information request message. |
| InfoQueryBinaryDataList | LIST(IQ\_BIN\_DATA) | (Optional) A list of TLV queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be processed by MIIS. See Table F. 15 for detailed definition. |
| InfoQueryRDFDataList | LIST(IQ\_RDF\_DATA) | (Optional) A list of RDF queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be processed by MIIS. |
| InfoQueryRDFSchemaURL | BOOLEAN | (Optional) A RDF Schema URL query. This field is required only when the value is “TRUE,” which indi­cates to query a list of RDF schema URLs. |
| InfoQueryRDFSchemaList | LIST(IQ\_RDF\_SCHM) | (Optional) A list of RDF schema queries. The order of the queries in the list identifies the priority of the query. The first query has the highest priority to be processed by MIIS. |
| MaxResponseSize | UNSIGNED\_INT(2) | (Optional) This field specifies the maximum size of Info Response parameters in MIS\_Get\_Information response primitive in octets. If this field is not speci­fied, the maximum size is set to 65 535. The actual maximum size forced by the IS server can be smaller than that specified by the IS client. |
| QuerierNetworkType | NETWORK\_TYPE | (Optional) The type of the network being used by the querier. This parameter is valid only with InfoQuery­BinaryDataList and InfoQueryRDFDataList. |
| UnauthenticatedInformation- Request | BOOLEAN | The value of UIR bit contained in the MIS\_Get\_Information request message received from the remote MISF. |

When generated

This primitive is generated by the MISF on receiving an MIS\_Get\_Information request message from a peer MISF. The order of the queries in each of InfoQueryBinaryDataList, InfoQueryRDFDataList, and InfoQueryRDFSchemaList parameters identifies the priority of the query. The first query has the highest priority to be processed by MIIS. Thus the order of the queries is maintained as indicated by the request message.

Effect on receipt

The recipient interprets the query request and retrieves the specified information. Once the information is retrieved, the recipient replies with the MIS\_Get\_Information.response primitive.

* + - * 1. MIS\_Get\_Information.response

Function

This primitive is used by an MIS user (i.e., MIIS Server) to respond to an MIS\_GET\_Information.indication primitive.

Semantics of service primitive

MIS\_Get\_Information.response (

DestinationIdentifier,  
Status,  
InfoResponseBinaryDataList,  
InfoResponseRDFDataList,  
InfoResponseRDFSchemaURLList,  
InfoResponseRDFSchemaList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | The local MISF or a remote MISF that will be the destination of this response. |
| Status | STATUS | Status of operation.  The response lists contains meaningful data if and only if the status is ‘0’. |
| InfoResponseBinaryDataList | LIST(IR\_BIN\_DATA) | (Optional) A list of TLV query responses. The list will be sorted from most preferred first to least preferred last. |
| InfoResponseRDFDataList | LIST(IR\_RDF\_DATA) | (Optional) A list of RDF query responses. The list will be sorted from most preferred first to least preferred last. |
| InfoResponseRDFSchemaURLList | LIST(IR\_SCHM\_URL) | (Optional) A list of RDF Schema URL. The list will be sorted from most preferred first to least preferred last. |
| InfoResponseRDFSchemaList | LIST(IR\_RDF\_SCHM) | (Optional) A list of RDF schema query responses. The list will be sorted from most preferred first to least preferred last. |

When generated

This primitive is generated by an MIS user in response to a received MIS\_Get\_Information.indication primitive. When the size of the Info Response parameters exceeds the maximum size specified in the MaxResponseSize parameter from MIS\_Get\_Information.indication primitive, one or more of the lower order list elements in Info Response parameters must be oMISed.

Effect on receipt

The recipient will return an MIS\_Get\_Information response message to the designated MIIS client.

* + - * 1. MIS\_Get\_Information.confirm

Function

This primitive is used by the MISF to respond to an MIS\_GET\_Information.request primitive.

Semantics of service primitive

MIS\_Get\_Information.confirm (

SourceIdentifier,  
Status,  
InfoResponseBinaryDataList,  
InfoResponseRDFDataList,  
InfoResponseRDFSchemaURLList,  
InfoResponseRDFSchemaList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | Specifies the MISF ID of the node that invoked MIS\_GET\_Information.response. |
| Status | STATUS | Status of operation.  The response lists contains meaningful data if and only if the status is ‘0’. |
| InfoResponseBinaryDataList | LIST(IR\_BIN\_DATA) | (Optional) A list of TLV query responses. The list will be sorted from most preferred first to least preferred last. |
| InfoResponseRDFDataList | LIST(IR\_RDF\_DATA) | (Optional) A list of RDF query responses. The list will be sorted from most preferred first to least preferred last. |
| InfoResponseRDFSchemaU- RLList | LIST(IR\_SCHM\_URL) | (Optional) A list of RDF Schema URL. The list will be sorted from most preferred first to least preferred last. |
| InfoResponseRDFSchemaList | LIST(IR\_RDF\_SCHM) | (Optional) A list of RDF schema query responses. The list will be sorted from most preferred first to least preferred last. |

When generated

This primitive is generated by the MISF on receiving an MIS\_Get\_Information Response message from a peer MISF.

Effect on receipt

The MIS user that requested the information utilizes the Info Response parameters and takes suitable action. However, if Status does not indicate “Success,” the recipient ignores any other returned values and, instead, performs appropriate error handling.

When the size of the Info Response parameters exceeds the maximum size specified in the MaxResponseSize parameter from MIS\_Get\_Information.request primitive, one or more of the lower order list elements in Info Response parameters must be oMISed.

MIS\_Push\_Information

* + - * 1. MIS\_Push\_Information.request

Function

This primitive is used by an MIS user (i.e., MIIS Server) to push information to the MN. MIS\_Push\_Information is generated by the MIIS Server to update policy information following a successful registration. This primitive can be generated at any time during the life time of the registration.

Semantics of service primitive

MIS\_Push\_Information.request (

DestinationIdentifier,  
InfoResponseBinaryDataList,  
InfoResponseRDFDataList,  
InfoResponseRDFSchemaURLList,  
InfoResponseRDFSchemaList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | The remote MISF, which is the destination of this request. |
| InfoResponseBinary- DataList | LIST(IR\_BIN\_DATA) | (Optional) A list of binary representations of Information Ele­ments. This list will be sorted from most preferred first to least preferred last. This list can include vendor specific IE’s for representing network policies. |
| InfoResponseRDF- DataList | LIST(IR\_RDF\_DATA) | (Optional) A list of network information in RDF. The list will be sorted from most preferred first to least preferred last. This list can include operator specific RDF data that can be used to represent operator policies. |
| InfoResponseRDF- SchemaURLList | LIST(IR\_SCHM\_URL ) | (Optional) A list of RDF Schema URL supported by the net­work. The list will be sorted from most preferred first to least preferred last. |
| InfoResponseRDF- SchemaList | LIST(IR\_RDF\_SCHM) | (Optional) A list of RDF Schema content supported by the network. The list will be sorted from most preferred first to least preferred last. |

At least one of the following parameters is specified: — InfoResponseBinaryDataList

* InfoResponseRDFDataList
* InfoResponseRDFSchemaURLList
* InfoResponseRDFSchemaList

When generated

This primitive is generated by the MIIS server to update any policies on the MN or to update any other IEs from the MIIS on to the MN.

Effect on receipt

The recipient will return an MIS\_Push\_Information indication message.

* + - * 1. MIS\_Push\_Information. indication

Function

This primitive is used by the MISF to notify MIS users of the information. This primitive is the result of receipt of an MIS\_Push\_information indication message from a remote MISF.

Semantics of service primitive

MIS\_Push\_Information.indication (

SourceIdentifier,  
InfoResponseBinaryDataList,  
InfoResponseRDFDataList,  
InfoResponseRDFSchemaURLList,  
InfoResponseRDFSchemaList  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | The remote MISF, which is the source of this request. |
| InfoResponseBinary- DataList | LIST(IR\_BIN\_DATA) | (Optional) A list of binary representations of Information Ele­ments. This list will be sorted from most preferred first to least preferred last. This list can include vendor specific IEs for representing network policies. |
| InfoResponseRDF- DataList | LIST(IR\_RDF\_DATA) | (Optional) A list of network information in RDF. The list will be sorted from most preferred first to least preferred last. This list can include operator specific RDF data that can be used to represent operator policies. |
| InfoResponseRDF- SchemaURLList | LIST(IR\_SCHM\_URL ) | (Optional) A list of RDF Schema URL supported by the net­work. The list will be sorted from most preferred first to least preferred last. |
| InfoResponseRDF- SchemaList | LIST(IR\_RDF\_SCHM) | (Optional) A list of RDF Schema content supported by the network. The list will be sorted from most preferred first to least preferred last. |

When generated

This primitive is generated by the MISF upon receiving an MIS\_Push\_Information indication message.

Effect on receipt

Upper layer entities take different actions upon notification.

**7.4.27 MIS\_Push\_Key**

**7.4.27.1 MIS\_Push\_key.request**

**7.4.27.1.1 Function**

This primitive is used to request a remote MISF (PoS) to install a key(s) in a target PoA(s).

**7.4.27.1.2 Semantics of service primitive**

MIS\_Push\_key.request (

DestinationIdentifier, LinkTupleIdentifierList

)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies a remote MISF that will be the destination of this request. |
| LinkTupleIdentifierList | LIST(LINK\_TUPLE\_ID) | This identifies a list of links of tar- get PoAs for which keys are pushed. |

**7.4.27.1.3 When generated**

This primitive is generated by an MIS user in the MN to request a remote MISF in the serving PoS to install a key in a target PoA.

**7.4.27.1.4 Effect on receipt**

The local MISF shall generate an MIS\_Push\_Key request message to the remote MISF.

**7.4.27.2 MIS\_Push\_key.indication**

**7.4.27.2.1 Function**

This primitive is used to pass a key to the corresponding MIS user on the serving PoS.

**7.4.27.2.2 Semantics of service primitive**

MIS\_Push\_key.indication ( SourceIdentifier, KeyMapping

) Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker, which is a remote MISF. |
| KeyMapping | KEY\_MAPPING | This specifies a mapping of a link identifier for which the key is pushed and a lifetime. |

**7.4.27.2.3 When generated**

This primitive is generated by the local MISF after receiving an MIS\_Push\_Key request message from the remote MISF.

**7.4.27.2.4 Effect on receipt**

A media specific key is delivered to the corresponding MIS user.

**7.4.27.3 MIS\_Push\_key.response**

**7.4.27.3.1 Function**

This primitive is used to indicate that the key installation request has been received and MIS user has executed it.

**7.4.27.3.2 Semantics of service primitive**

MIS\_Push\_key.response (

DestinationIdentifier, LinkTupleIdentifierList, Status

)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies a remote MISF that will be the destination of this response. |
| LinkTupleIdentifierList | LIST(LINK\_TUPLE\_ID) | This identifies a list of links for which keys are pushed. |
| Status | STATUS | This represents the operation result. |

**7.4.27.3.3 When generated**

This primitive is generated by an MIS user after receiving an MIS\_Push\_Key.indication primitive.

**7.4.27.3.4 Effect on receipt**

The local MISF shall generate an MIS\_Push\_Key response message to the remote MISF.

**7.4.27.4 MIS\_Push\_Key.confirm**

**7.4.27.4.1 Function**

This primitive is used to notify the MIS user (in MN side) about the status of the requested operation.

**7.4.27.4.2 Semantics of service primitive**

MIS\_Push\_key.confirm (

SourceIdentifier, LinkTupleIdentifierList, Status

)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker, which is a remote MISF. |
| LinkTupleIdentifierList | LIST(LINK\_TUPLE\_ID) | This identifies a list of links for which keys are pushed. |
| Status | STATUS | This represents the operation result. |

**7.4.27.4.3 When generated**

This primitive is generated after receiving an MIS\_Push\_Key response message.

**7.4.27.4.4 Effect on receipt**

A media specific key must be installed in the link layer.

**7.4.28 MIS\_LL\_Auth**

The primitives defined are to carry out a proactive authentication over MIS between the MN and the PoS using link layer frames. The authentication is conducted with the media specific authenticator that serves the target PoA.

**7.4.28.1 MIS\_LL\_Auth.request**

**7.4.28.1.1 Function**

This primitive carries link-layer frames for authentication purposes.

**7.4.28.1.2 Semantics of service primitive**

MIS\_LL\_Auth.request ( DestinationIdentifier, LinkIdentifier, LLInformation

) Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies a remote MISF that will be the destination of this request. |
| LinkIdentifier | LINK\_TUPLE\_ID | This identifies a PoA that is also the authenticator. |
| LLInformation | LL\_FRAMES | This carries link layer frames. |

**7.4.28.1.3 When generated**

This primitive is generated by an MIS user to start an authentication process based on link-layer frames.

**7.4.28.1.4 Effect on receipt**

The local MISF shall generate an MIS\_LL\_Auth request message to the remote MISF.

**7.4.28.2 MIS\_LL\_Auth.indication**

**7.4.28.2.1 Function**

This primitive is used by the remote MISF to notify the corresponding MIS user about the reception of an

MIS\_LL\_Auth request message.

**7.4.28.2.2 Semantics of service primitive**

MIS\_LL\_Auth.indication ( SourceIdentifier, LinkIdentifier, LLInformation

) Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker, which is a remote MISF. |
| LinkIdentifier | LINK\_TUPLE\_ID | This identifies a PoA that is also the authenticator. |
| LLInformation | LL\_FRAMES | This carries link layer frames. |

**7.4.28.2.3 When generated**

This primitive is generated by a remote MISF after receiving an MIS\_LL\_Auth request message.

**7.4.28.2.4 Effect on receipt**

The MIS user must generate an MIS\_LL\_Auth.response primitive.

**7.4.28.3 MIS\_LL\_Auth.response**

**7.4.28.3.1 Function**

This primitive is used by an MIS user to provide the link-layer frames to the local MISF.

**7.4.28.3.2 Semantics of service primitive**

MIS\_LL\_Auth.response (

DestinationIdentifier, LinkIdentifier, LLInformation,

Status

)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| DestinationIdentifier | MISF\_ID | This identifies a remote MISF that will be the destination of this response. |
| LinkIdentifier | LINK\_TUPLE\_ID | This identifies a PoA that is also the authenticator. |
| LLInformation | LL\_FRAMES | This carries link layer frames. |
| Status | STATUS | Status of the authentication. |

**7.4.28.3.3 When generated**

This primitive is generated after receiving an MIS\_LL\_Auth.indication primitive.

**7.4.28.3.4 Effect on receipt**

The local MISF must generate an MIS\_LL\_Auth response message in order to provide the required infor- mation until the authentication is finished.

**7.4.28.4 MIS\_LL\_Auth.confirm**

**7.4.28.4.1 Function**

This primitive is used to notify the corresponding MIS user about the reception of an MIS\_LL\_Auth response message.

**7.4.28.4.2 Semantics of service primitive**

MIS\_LL\_Auth.confirm (SourceIdentifier

LLidentifier

Status)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| SourceIdentifier | MISF\_ID | This identifies the invoker, which is a remote MISF. |
| LLInformation | LL\_FRAMES | This carries link layer frames. |
| Status | STATUS | Status of the authentication. |

**7.4.28.4.3 When generated**

This primitive is generated by the remote MISF after receiving an MIS\_LL\_Auth response message.

**7.4.28.4.4 Effect on receipt**

The MIS user may generate an MIS\_LL\_Auth.request primitive unless the authentication is completed.

* 1. MIS\_NET\_SAP primitives

MIS\_TP\_Data

The primitives associated with data transfers are as follows:

* MIS\_TP\_Data.request
* MIS\_TP\_Data.indication
* MIS\_TP\_Data.confirm

The MISF uses the MIS\_TP\_Data.request primitive to request that an MIS PDU be transported. The transport service provider uses the MIS\_TP\_Data.indication primitive to indicate the arrival of an MIS PDU. MIS\_TP\_Data.confirm primitive is used to acknowledge the successful transfer of the MIS PDU.

* + - * 1. MIS\_TP\_Data.request

Function

This primitive is the request for transfer of an MIS PDU.

Semantics

MIS\_TP\_Data.request (

TransportType,  
SourceAddress,  
DestinationAddress,  
ReliableDeliveryFlag,  
MISProtocolPDU  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| TransportType | TRANSPORT\_TYPE | Identifies the protocol layer specific transport option. |
| SourceAddress | TRANSPORT\_ADDR | Protocol layer specific Transport Address of entity that has the Source MISF. |
| DestinationAddress | TRANSPORT\_ADDR | Protocol layer specific Transport Address of entity that has the Destination MISF. |
| ReliableDeliveryFlag | BOOLEAN | Indicate that the data is sent reliably and an error is gener­ated if delivery fails.  True: Reliable delivery is required.  False: Reliable delivery is not required. |
| MISProtocolPDU | OCTET\_STRING | MIS Protocol PDU to be transferred. |

When generated

This primitive is used to request that an MIS PDU be transported to a remote MISF.

Effect on receipt

The receipt of this primitive causes the selected transport service provider to attempt to transport the MIS PDU.

* + - * 1. MIS\_TP\_Data.indication

Function

This primitive is the indication of a received MIS PDU.

Semantics

MIS\_TP\_Data.indication (

TransportType,  
SourceAddress,  
DestinationAddress,  
ReliableDeliveryFlag,  
MISProtocolPDU  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| TransportType | TRANSPORT\_TYPE | Identifies the protocol layer specific transport option. |
| SourceAddress | TRANSPORT\_ADDR | Protocol layer specific Transport Address of entity that has the Source MISF. |
| DestinationAddress | TRANSPORT\_ADDR | Protocol layer specific Transport Address of entity that has the Destination MISF. |
| ReliableDeliveryFlag | BOOLEAN | Indicate that the data is sent reliably and an error generated if delivery fails.True: Reliable delivery is require; False: Reliable delivery is not required. |
| MISProtocolPDU | OCTET\_STRING | MIS Protocol PDU received. |

When generated

This primitive is used by the transport service provider to indicate that an MIS PDU has been received from a remote MISF.

Effect on receipt

The receipt of this primitive causes the MISF to receive the MIS PDU that was transported.

* + - * 1. MIS\_TP\_Data.confirm

Function

This primitive is used to confirm an acknowledged transfer.

Semantics

MIS\_TP\_Data.confirm (

Status,  
TransportType,  
SourceAddress,  
DestinationAddress  
)

Parameters:

|  |  |  |
| --- | --- | --- |
| **Name** | **Data type** | **Description** |
| Status | STATUS | Status of operation. |
| TransportType | TRANSPORT\_TYPE | Identifies the protocol layer specific transport option. |
| SourceAddress | TRANSPORT\_ADDR | Protocol layer specific Transport Address of entity that has the Source MISF. |
| DestinationAddress | TRANSPORT\_ADDR | Protocol layer specific Transport Address of entity that has the Destination MISF. |

When generated

This primitive is passed from the transport service provider to the MISF to confirm that a request to transfer an MIS PDU succeeded.

Effect on receipt

Upon receipt of this primitive, the receiving MISF stops its retransmission timer for the corresponding request. When the MISF does not receive this primitive for a pre-defined time after transmitting an MIS\_TP\_Data.request with ReliableDeliveryFlag set to TRUE, the MISF attempts to retransmit the MIS\_TP\_Data.request.

1. Media independent services protocol
   1. Introduction

The MIS Function entities in MN and network entities communicate with each other using the MIS protocol messages specified in this clause. The MIS protocol defines message formats for exchanging these messages between peer MIS Function entities. These messages are based on the primitives that are part of the MIS Services.

* 1. MIS protocol description

MIS protocol transaction

The media independent services protocol defines a message exchange between two MISF entities to support remote MISF services. An MIS transaction is identified by a sequence of messages with the same Transaction-ID submitted to, or received from, one specific remote MISF ID.

At any given moment, an MIS node shall have no more than one transaction pending for each direction with a certain MIS peer. In other words, the MIS node shall wait until any pending outgoing transaction is completed before it creates another outgoing transaction for the same peer. Similarly, the MIS node shall wait until any pending incoming transaction is completed before it creates another incoming transaction for the same peer.

MIS protocol acknowledgement service

The acknowledgement service shall be used when the MIS transport used for remote communication does not provide reliable services. When the MIS transport is reliable, the use of the acknowledgement service is not needed. The acknowledgement service is particularly useful when the underlying transport used for remote communication does not provide reliable services. When the MIS transport is reliable, the acknowledgement service is optional.

The source MISF requests for an acknowledgement message to ensure successful receipt of an MIS protocol message. This MIS message is used to acknowledge the successful receipt of an MIS protocol message at the destination MISF.

The MIS acknowledgement service is supported by the use of two bits of information that are defined exclusively for acknowledgement (ACK) usage in the MIS header. The ACK-Req bit is set by the source MIS node and the ACK-Rsp bit is set by the destination MIS node to utilize the acknowledgement service. It is expected that the underlying transport layer would take care of ensuring the integrity of the MIS protocol message during delivery.

When seeking acknowledgement service, the source MIS node shall start a retransmission timer after sending an MIS protocol message with the ACK-Req bit set and saves a copy of the MIS protocol message while the timer is active. The algorithm defined in IETF RFC 2988 is used to calculate the value of the retransmission timer. If the acknowledgement message is not received before the expiration of the timer, the source MIS node immediately retransmits the saved message with the same Message-ID and with the same Transaction-ID (with ACK-Req bit set). If the source MIS node receives the acknowledgement before the expiration of the timer on the first or any subsequent retransmitted attempt, then the source MIS node has ensured the receipt of the MIS packet and therefore, resets the timer and releases the saved copy of the MIS protocol message. During retransmission, if the source MIS node receives the acknowledgement for any of the previous transmission attempts then the source MIS node determines successful delivery of the message and does not have to wait for any further acknowledgements for the current message. The source MIS node retransmits an MIS protocol message with ACK-Req bit set until it receives an acknowledgment or the number of retransmissions reaches its maximum value. The maximum number of retransmissions can be configured through a parameter defined in the MIB (see Annex J). The source MIS node does not attempt to retransmit a message with same Message-ID and Transaction-ID when the ACK-Req bit was not set in the first MIS message. Implementations may consider adjusting the retransmission time-out (RTO) when operating over links with MNs that may be sleeping.

When a destination MIS node receives an MIS protocol message with the ACK-Req bit set, then the destination MIS node returns an MIS message with the ACK-Rsp bit set and copying the Message-ID and Transaction-ID from the received MIS protocol message. The MIS message with the ACK-Rsp bit set has only the MIS header and no other payload. In instances where the destination MIS node immediately processes the received MIS protocol message and a response is immediately available, then the ACK-Rsp bit is set in the corresponding MIS protocol response message.

The destination MIS node responds with an acknowledgement message for duplicate MIS messages (messages with same transaction-ID) that have the ACK-Req bit set. However, the destination MIS node does not process these duplicate messages if it has already done so. If a destination MIS node receives an MIS protocol message with no ACK-Req bit set, then no action is taken with respect to the acknowledgement service.

In all cases, the MIS protocol message in a transaction is processed only once at the destination MIS node, irrespective of the number of received messages with the ACK-Req bit set. The destination MIS node sets the ACK-Rsp bit in an MIS protocol response message and additionally requests acknowledgement by setting the ACK-Req bit for the same MIS protocol response message.

MIS protocol transaction state diagram

* + - * 1. State machines

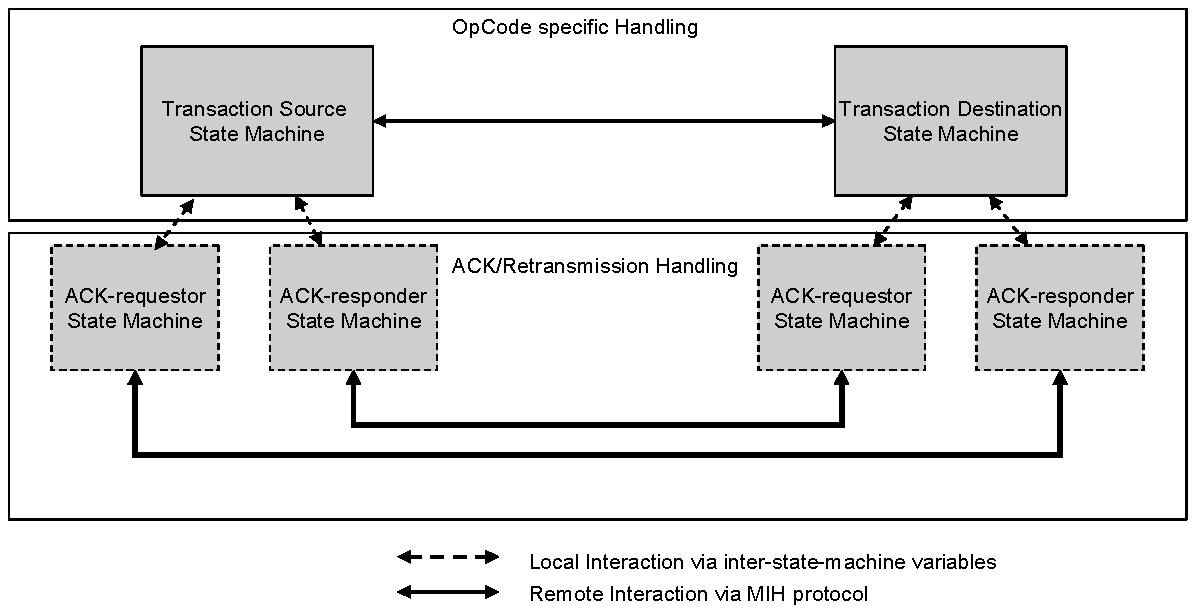
A node that has a new available message to send related to a new transaction is called transaction source and starts the transaction source state machine. In the same manner, a node that receives a message related to a new transaction is called transaction destination node and starts the destination transaction state machine.

If the ACK feature is being used by the source and/or destination transaction node, the ACK-Requestor and/ or ACK-Responder state machine is started (specific conditions specified below). The ACK related state machine is run in parallel to the transaction source/destination state machines.

Each transaction is represented in an MISF by an instance of the transaction source or destination state machine. Optionally, each transaction can also have one instance of ACK-Requestor or one instance of ACK-Responder state machine, or both.

All instances of the state machines related to one transaction have access to inter-state-machine variables, constants and procedures, which are not accessible by the state machines related to other transactions. The inter-state-machine variables allow communications between state machines for a given transaction. There are no cases where two or more state machines for a given transaction write the same inter-state-machine variable at the same time. Intra-state-machine variables, constants, and procedures can only be accessed within a single state machine for a given transaction.

Figure 21 illustrates the interaction of transaction source/destination state machines with the ACK related state machines.



**Figure 21—State machines interactions**

* + - * 1. Notational conventions used in state diagrams

State diagrams are used to represent the operation of an MIS transaction as a group of connected, mutually exclusive states. At any given time, only one state of each state machine can be active per transaction instance.

Each state is represented in the state diagram as a rectangular box, divided into two parts by a horizontal line. The upper part contains the state identifier, written in uppercase letters. The lower part contains any procedures that are executed on entry to the state.

All permissible transitions between states are represented by arrows, the arrowhead denoting the direction of the possible transition. Labels attached to arrows denote the condition(s) that shall be met in order for the transition to take place.

A transition that is global in nature (i.e., a transition that occurs from any of the possible states if the condition attached to the arrow is met) is denoted by an open arrow; i.e., no specific state is identified as the origin of the transition.

On entry to a state, the procedures defined for the state (if any) are executed exactly once, in the order that they appear on the page. Each action is deemed to be atomic; i.e., execution of a procedure completes before the next sequential procedure starts to execute. No procedures execute outside of a state block. On completion of all of the procedures within a state, all exit conditions for the state (including all conditions associated with global transitions) are evaluated continuously until such a time as one of the conditions is met. All exit conditions are regarded as Boolean expressions that evaluate to TRUE or FALSE; if a condition evaluates to TRUE, then the condition is met.

The label UCT denotes an unconditional transition (i.e., UCT always evaluates to TRUE).

A variable that is set to a particular value in a state block retains this value until a subsequent state block executes a procedure that modifies the value.

Should a conflict exist between the interpretation of a state diagram and either the corresponding transition tables or the textual description associated with the state machine, the state diagram takes precedence.

The interpretation of the special symbols and operators used in the state diagrams is defined in Table 18;  
these symbols and operators are derived from the notation of the “C” programming language, ANSI X3.159.

**Table 18—State machine symbols**

|  |  |
| --- | --- |
| **Symbol** | **Interpretation** |
| ( ) | Used to force the precedence of operators in Boolean expressions and to delimit the argument(s) of actions within state boxes. |
| ; | Used as a terminating delimiter for actions within state boxes. Where a state box  contains multiple actions, the order of execution follows the normal English language conven­tions for reading text. |
| = | Assignment action. The value of the expression to the right of the operator is assigned to the vari­able to the left of the operator. Where this operator is used to define multiple assignments, (e.g., a = b = X) the action causes the value of the expression following the right-most assignment oper­ator to be assigned to all of the variables that appear to the left of the right-most assignment oper­ator. |
| ! | Logical NOT operator. |
| && | Logical AND operator. |
| || | Logical OR operator. |
| (statement1 ? statement2: statement3) | Conditional action. If statement1 evaluates to TRUE, then statement2 is executed. Otherwise statement3 is executed. |
| == | Equality. Evaluates to TRUE if the expression to the left of the operator is equal in value to the expression to the right. |
| < | Less than. Evaluates to TRUE if the value of the expression to the left of the operator is less than the value of the expression to the right. |
| ++ | Arithmetic increment by one operator. |

* + - * 1. Inter-state-machine variables

Inter-state-machine variables are available for use by more than one state machine related to one transaction instance and are used to perform inter-state-machine communication and initialization functions within that transaction.

Exported variables are inter-state-machine variables that are also readable and writable from entities external to the state machines. The inter-state-machine and exported state machine variables are specified in Table 19 and Table 20, respectively.

**Table 19—Inter-state-machine variables**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Description** |
| Opcode | OPCODE | An Opcode. |
| MID | MID | A message identifier. |

**Table 19—Inter-state-machine variables *(continued)***

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Description** |
| AckRequestorStatus | ENUMERATED | Indicates the status of the ACK requestor state machine. This variable is initialized by the transac­tion source state machine or transaction destination state machine and changed by the ACK requestor  state machine. The following values are valid:  1 ONGOING  2 SUCCESS  3 FAILURE |
| TransactionStopWhen | UNSIGNED\_INT(1) | A timer to stop the transaction. |
| RetransmissionWhen | UNSIGNED\_INT(1) | A timer to retransmit a message. |

**Table 20—Exported state machine variables**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Description** |
| TID | TID | A transaction identifier. |
| MyMisfID | MISF\_ID | The MISF ID of this MIS node. |
| PeerMisfID | MISF\_ID | The MISF ID of the peer MIS node. |
| MsgIn | MIS\_MESSAGE | A valid incoming message received from a remote MISF. An incoming message is valid in terms of state machine operation if the message has the Operation Code of the value Request (0x1), Response (0x2), or Indication (0x3). |
| MsgInAvail | BOOLEAN | This variable is set to TRUE by an entity external to the state machines when a valid incoming message is available for a transaction. The transaction corresponds to an instance of either Transaction Source State Machine or Transaction Destination State Machine depending on the Operation Code, Destination Identifier TLV, and ACK-Rsp bit of the message as shown in Table 21. The correspon­dence between an incoming message and a transaction is based on TID, MyMisfID, and PeerMisfID variables of Transaction Source or Destination State Machine against the Transaction ID field, Destination Identifier TLV, and Source Identifier TLV of the incoming message, respec­tively. This variable is initialized to FALSE by the external entity. This variable is set to FALSE by the state machines once the incoming message has been processed. It is the responsibility of the external entity to set this variable to TRUE such that this MIS node has no more than one trans­action pending for each direction with a certain MIS peer. |
| MsgOut | MIS\_MES SAGE | A valid outgoing message generated by the local MISF to be sent to the remote MISF. An outgoing message is valid in terms of state machine operation if the message has the Operation Code of the value Request (0x1), Response  (0x2), or Indication (0x3). |

**Table 20—Exported state machine variables *(continued)***

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Description** |
| MsgOutAvail | BOOLEAN | This variable is set to TRUE by an entity external to the state machines or by Transaction Source or Destination State Machine when a valid outgoing message is available for a transaction. The transaction corresponds to an instance of either Transaction Source State Machine or Transaction Destination State Machine depending on the Operation Code and Destination Identifier TLV of the mes­sage as shown in Table 22. The correspondence between an outgoing message and a transaction is made based on matching TID, MyMisfID, and PeerMisfID variables of Transaction Source or Destination State Machine instances against the Transaction ID field, Source Identifier TLV, and Destination Identifier TLV of the outgoing message, respectively. This variable is initialized to FALSE by the external entity. It is the responsibility of the external entity to set this variable to TRUE such that this MIS node has no more than one transaction pending for each direction with a certain MIS peer. |
| TransactionStatus | ENUMERATED | Indicates the status of the transaction. This variable is writ­ten by the state machine and read by the MISF.  The following values are valid:  1 ONGOING  2 SUCCESS  3 FAILURE |
| StartAckRequestor | BOOLEAN | This variable is initialized to FALSE by an external entity. The instance of ACK-requestor state machine is started when this variable is set to TRUE by its associated transac­tion source or destination state machine. |
| StartAckResponder | BOOLEAN | This variable is initialized to FALSE by an external entity. The instance of ACK-responder state machine is started when this variable is set to TRUE by its associated transac­tion source or destination state machine. |

**Table 21—State Machines to be searched for incoming message**

|  |  |  |  |
| --- | --- | --- | --- |
| **Operation code** | **ACK-Rsp bit** | **Contains MIS service specific TLVs or a fragment payload** | **State machine instances to be searched: transaction *source* state machine (S) or transaction destination state machine (D)** |
| Request (0x1)/Indication (0x3) | 0 | — | D |
| 1 | — | S |
| Response (0x2) | 0 | — | S |
| 1 | Yes | S |
| No | D |

**Table 22—State Machines to be searched for outgoing message**

|  |  |
| --- | --- |
| **Operation code** | **State machine instances to be searched: transaction source state machine (S) or transaction destination state machine (D)** |
| Request (0x 1) / Indication (0x3) | S |
| Response (0x2) | D |

* + - * 1. Inter-state-machine procedures

1. **BOOLEAN Process(MIS\_MESSAGE)**—This procedure processes the incoming message passed as an input variable. A value of TRUE is returned if an outgoing message is available in response to the incoming message. Otherwise, a value of FALSE is returned.
2. **void Transmit(MIS\_MESSAGE)**—This procedure transmits the message passed as the input variable.
3. **BOOLEAN IsMulticastMsg(MIS\_MESSAGE)**—This procedure outputs TRUE if the input message has a multicast destination MISF\_ID. Otherwise, it outputs FALSE.
4. **MISF\_ID SrcMISF\_ID(MIS\_MESSAGE)**—This procedure obtains a Source Identifier TLV from the message passed as the input and returns the value of the TLV.
5. **MISF\_ID DstMISF\_ID(MIS\_MESSAGE)**—This procedure obtains a Destination Identifier TLV from the message passed as the input and returns the value of the TLV.
6. **void SetMISF\_ID(MIS\_MESSAGE, MISF\_ID, MISF\_ID)**—This procedure inserts a Source Identifier TLV and a Destination Identifier TLV into the MIS message. The first MISF\_ID is used as the value of the Source Identifier TLV. The second MISF\_ID is used as the value of the Destination Identifier TLV.
   * + - 1. Inter-state-machine constants
7. **TransactionLifetime**—The maximum time from the initiation of a transaction until its termination.
8. **Request**—An OPCODE value of 0x1.
9. **Response**—An OPCODE value of 0x2.
10. **Indication**—An OPCODE value of 0x3.
    * + - 1. Timers

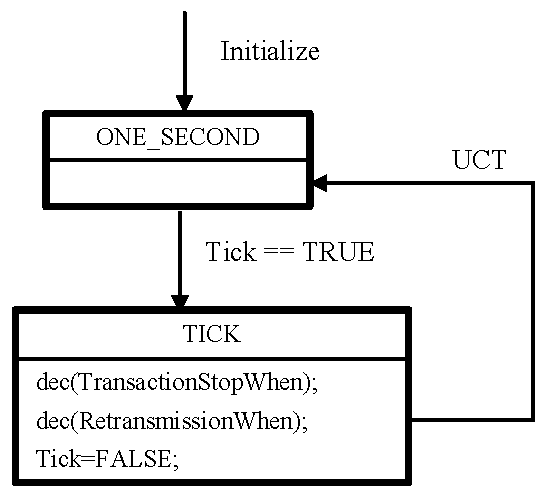
The timers defined for these state machines are decremented, if their value is non-zero, by the operation of Transaction Timers state machine. All timers have a resolution of one second, i.e., the initial values used to start the timers are integer values, and they represent the timer period as an integral number of seconds.

Intra-state-machine variables and constants

1. **Tick**—This variable is set in response to a regular one-second tick generated by an external system clock function. Whenever the system clock generates a one-second tick, the tick variable is set to TRUE. The variable is set to FALSE by the operation of the state machine. The operation of the sys­tem clock functions is not otherwise specified by the standard.
2. **void dec(Timer)**—This procedure decrements the timer only if its value is greater than 0.

Transaction timers state machine

The transaction timers state machine (see Figure 22) for a given transaction is responsible for decrementing the timer variables for this transaction each second, in response to an external system clock function. The timer variables are used, and set to their initial values, by the operation of the individual state machines for the transaction.



**Figure 22—Transaction timers state machine**

* + - * 1. Transaction source and destination state machines

Intra-state-machine variables

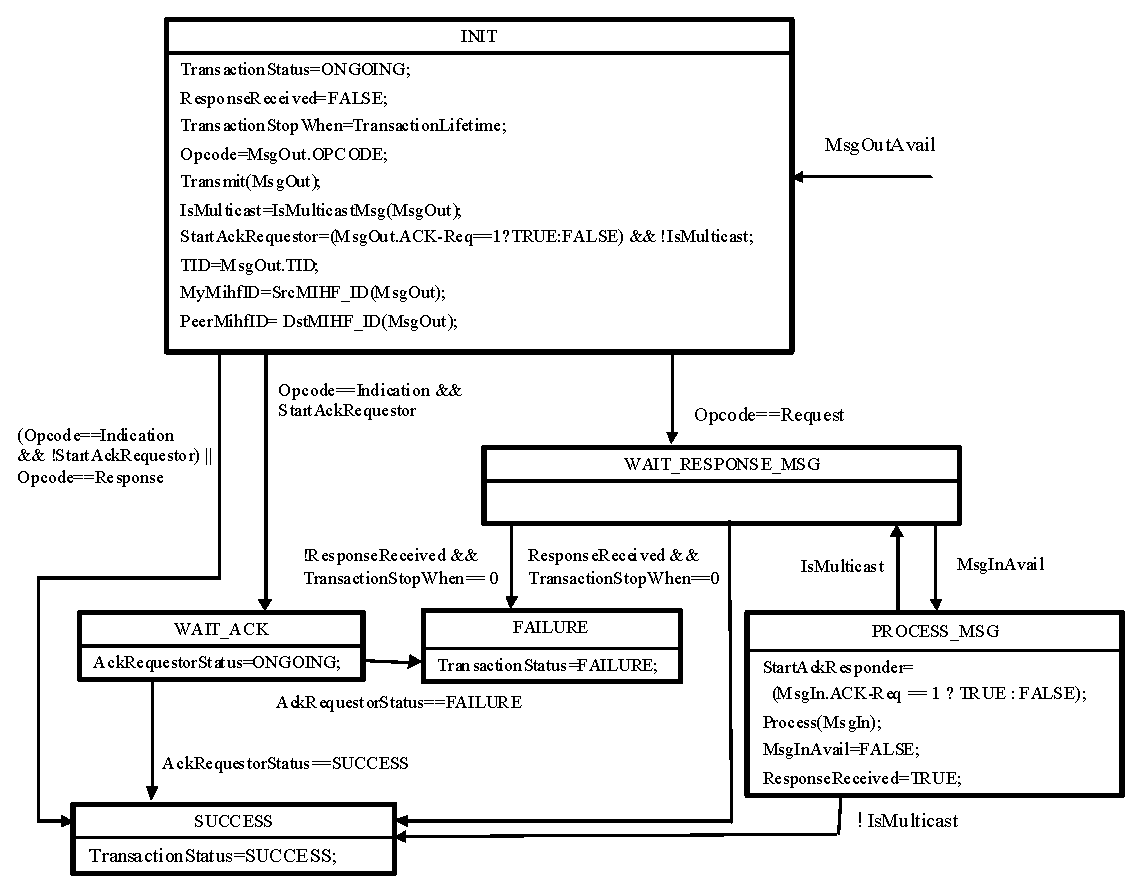
1. **IsMulticast**—This variable’s type is BOOLEAN. When its value is TRUE, it indicates that a mes­sage has a multicast destination MISF\_ID. Otherwise, its value is FALSE.

Intra-state-machine procedures

1. **ResponseReceived**—This variable’s type is BOOLEAN. When its value is TRUE it indicates that a Response message has been received. Otherwise, its value is FALSE.
2. **TID NewTID(void)**—This procedure generates a new transaction ID for the transaction generated by the new available message.

Transaction source state machine

The transaction source state machine (see Figure 23) is started, and related transaction initiated, when a message related to a new transaction is available to be sent (MsgOutAvail is TRUE). The transaction terminates when it transits to the SUCCESS state and any ACK related state machines if started were terminated; or if it transits to the FAILURE state. An instance of transaction source state machine can cease to exist once the value of TransactionStatus is set to either SUCCESS or FAILURE.

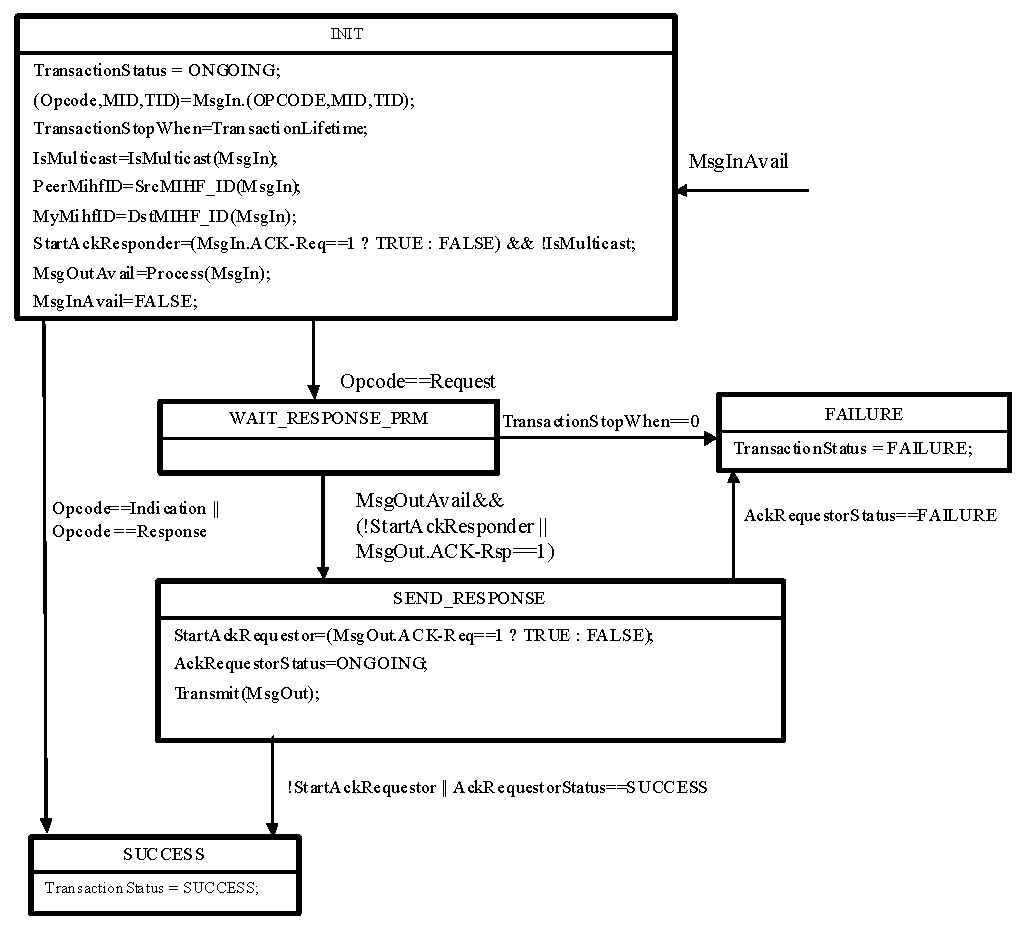


**Figure 23—Transaction source state machine**

Transaction destination state machine

The transaction destination state machine (see Figure 24) is started, and related transaction initiated, when a message related to a new transaction is received (MsgInAvail is TRUE).

The transaction terminates when it transits to the FAILURE state or SUCCESS state and any ACK related state machines, if started, were terminated. An instance of transaction destination state machine can cease to exist once the value of TransactionStatus is set to either SUCCESS or FAILURE.



**Figure 24—Transaction destination state machine**

* + - * 1. ACK related state machines

The ACK-requestor state machine is started when the StartAckRequest variable turns TRUE and ACK­responder state machine is started when StartAckResponder variable turns TRUE.

Intra-state-machine variables

1. **DUP**—This variable is of type MIS\_MES SAGE and represents an MIS message that has already been sent. This variable is used within ACK Responder state machine.
2. **ACK**—This variable is of type MIS\_MESSAGE and represents an MIS message with the ACK­Rsp bit set and the same message ID and transaction ID as the MIS message it acknowledges. This variable is used within ACK Responder state machine.
3. **RtxCtr**—This variable is of type UNSIGNED\_INT(1) and represents a number of retransmissions of a specific message. This variable is used within ACK Requestor state machine.

Intra-state-machine constants

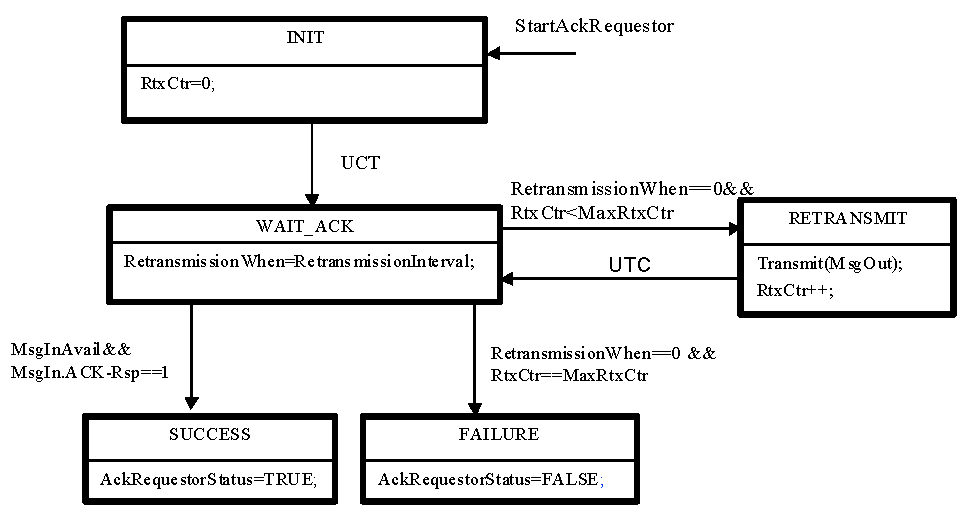
1. **RetransmissionInterval**—The time interval between two subsequent transmissions of a specific message.
2. **MaxRtxCtr**—The maximum number of times that a message will be retransmitted, if retransmis­sion conditions occur.

The maximum number of retransmissions and the retransmission interval depends on the characteristics of the underlying transport. These configuration parameters are defined in a MIB, see Annex J.

Note that the maximum number of retransmission is bounded by the transaction lifetime.

ACK requestor state machine

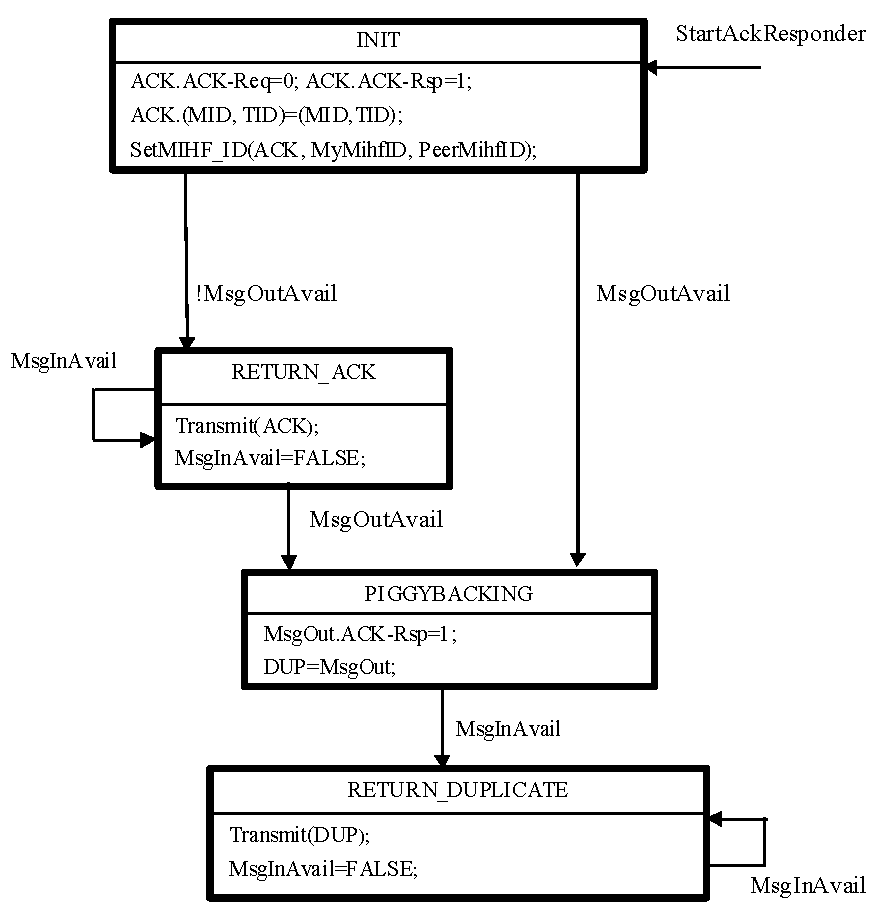
The ACK requestor state machine (see Figure 25) is started when the StartAckRequestor variable turns to TRUE in a source or destination transaction state machine. This state machine uses the inter-state-machine variables set by the originating state machine. This state machine terminates when it transits to the FAILURE state or SUCCESS state. An instance of ACK requestor state machine can cease to exist once the AckRequestorStatus is set to either SUCCESS or FAILURE state or its associated transaction source or transaction destination state machine ceases to exist.



**Figure 25—ACK requestor state machine**

ACK responder state machine

The ACK responder state machine (see Figure 26) is started when the StartAckResponder variable turns to TRUE in a source or destination transaction state machine. This state machine uses the inter-state-machine variables set by the originating state machine. An instance of ACK responder state machine can cease to exist once its associated transaction source or transaction destination state machine ceases to exist.



**Figure 26—ACK responder state machine**

Other considerations

* + - * 1. Congestion control and load management

The MIS protocol does not provide direct support for congestion control. Therefore, it is recommended to run the MIS protocol over congestion aware transport layers.

In order to help prevent congestion, flow control mechanisms are implemented at the MISF. A single rate limiter applies to all traffic (for all interfaces and message types). It applies to retransmissions, as well as new messages, although an implementation can choose to prioritize one over the other. When the rate limiter is in effect, MIS messages are queued until transmission is re-enabled, or an error condition is indicated back to local upper layer applications. The rate limiting mechanism is implementation specific, but it is recommended that a token bucket limiter as described in IETF RFC 4443 [B26] be used.

When an MISF suffers from overload, it drops requests from MIS requestors. For example, messages could be dropped from a particular requestor if that requestor could be established as the origin of a denial of service attack. Any reliable delivery function indicates a flow control back to the requestor, and an MISF invokes flow control towards a specific requestor when overloaded with reliably delivered messages.

* + - * 1. Reliability

MIS protocol messages are delivered via media dependent transport. To ensure proper operation, a reliable message delivery service is required. If the media dependent transport is unreliable, then the Acknowledgement Service shall be enabled, as specified in 8.2.2. If the media dependent transport is reliable, then the Acknowledgement Service may be implemented.

A reliable media dependent transport is one that exhibits a message loss rate of less than 0.01%.

* + - * 1. MISF discovery

General

The MISF discovery refers to the procedure that allows one MISF to discover its peer MISFs (e.g., an MN discovers available peer MISFs in an access network). MISF discovery can be done either at layer 2 or layer 3. MISF discovery at L2 is performed either in media-specific manner (e.g., using IEEE 802.11 Beacon frames, IEEE 802.16 DCD) or using multicast data frames as described in 8.2.4.3.2 and 8.2.4.3.3. MISF discovery mechanisms at layer 3 are defined in the IETF drafts [B29], [B30], and [B3 1].

Combined MIS function discovery and capability discovery over data plane

Combined MIS function discovery and capability discovery is performed to discover the MISF ID, the peer MISF transport address, and MISF capabilities at the same time. As stated in 6.2.3, MISF Discovery can be implicitly performed using the MIS capability discovery when both MIS nodes are residing in the same multicast domain (where an MIS node’s multicast data frame can be delivered using a group MAC address). If MISF ID and transport address are known (e.g., pre-configured) MISF uses MIS\_Capability\_Discover messages to discover MISF capabilities only. The following subclauses refer to the MIS capability discovery both as a means to discover the MISF and its capabilities.

Unsolicited MIS capability discovery

An MISF discovers peer MISF entities and their capabilities by listening to media-specific broadcast control messages. For example, by listening to a media-specific broadcast message such as a Beacon frame in IEEE Std 802.11 or a DCD in IEEE Std 802.16, link layers on an MN can then forward the detected MIS capabilities to its MISF.

Solicited MIS capability discovery

An MISF (the requestor) discovers its peer MIS functions and capabilities by multicasting or unicasting an MIS\_Capability\_Discover request message to either its multicast domain or a known MISF ID, respectively. Only MIS network entities respond to a multicast MIS\_Capability\_Discover request.

When a peer MIS function (the responder) receives the MIS\_Capability\_Discover request message, it sends MIS\_Capability\_Discover response message back to the requestor. The response is sent by using the same transport type over which the request message was received. When the requestor receives the unicast MIS\_Capability\_Discover response message, it learns the responder’s MISF ID by checking the source ID of MIS\_Capability\_Discover response.

For complete operation, the requestor sets a timer at the time of sending an MIS\_Capability\_Discover request during which time the requestor is in waiting state for a response from the responder. When the response message is received while the timer is running, the requestor stops the timer and finishes the MIS function and capability discovery procedure. When the timer expires without receiving a response message, the requestor tries the combined MIS function discovery and capability discovery procedure by using a different transport or terminates the MIS function and capability discovery procedure.

If the MIS capability discovery is invoked upon receiving MIS capability advertisement in unauthenticated state through media specific broadcast messages, such as beacon frames and DCD, destination MISF ID is filled with multicast MISF ID and this message is transmitted over the control plane using an L2 management frame, such as an IEEE 802.11 management action frame or an IEEE 802.16 MAC management message. This message contains the SupportedMisEventList, SupportedMisCommandList, SupportedISQueryTypeList, SupportedTransportList, and MBBHandoverSupport TLVs to enable the receiving MISF to discover the sending MISF’s capability. Therefore, peer MISF entities can discover each other’s MIS capability by one MIS protocol message transaction. When the requestor receives the unicast MIS\_Capability\_Discover response message, which is embedded in the media specific control message, it retrieves the responder’s MISF ID by checking the source of the MIS\_Capability\_Discover response message.

* 1. MIS protocol identifiers

The following identifiers are used in MIS protocol messages:

* MISF ID
* Transaction ID

MISF ID

MISF Identifier (MISF ID) is an identifier that is required to uniquely identify an MISF entity for delivering the MIS services. MISF ID is used in all MIS protocol messages. This enables the MIS protocol to be transport agnostic.

MISF ID is assigned to the MISF during its configuration process. The configuration process is outside the scope of the standard.

Multicast MISF ID is defined as an MISF ID of zero length. A multicast MISF ID can be used when destination MISF ID is not known to a source MISF. The MISF ID is of type MISF\_ID. (See F.3.11.)

Transaction ID

Transaction Identifier (Transaction ID) is an identifier that is used to match a request message with its corresponding response message. This identifier is also required to match each request, response or indication message and its corresponding acknowledgment. This identifier is created at the node initiating the transaction and it is carried over within the fixed header part of the MIS protocol frame.

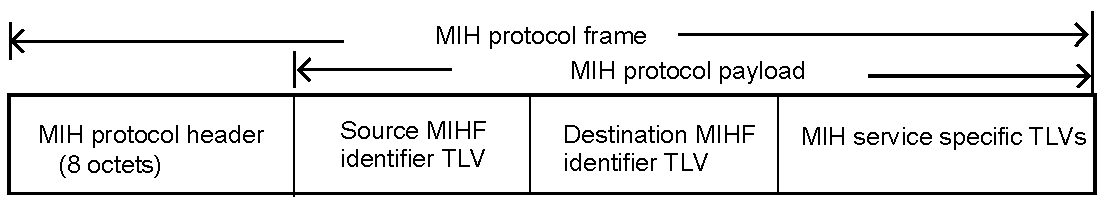
Transaction ID is defined as a 16 bit long unsigned integer whose value is unique among all the pending transactions between a given pair of the sender and receiver. For example, this could be an integer that starts from a random initial value and incremented by one (modulo 216) every time a new Transaction ID is generated.

* 1. MIS protocol frame format

General frame format

In MIS protocol messages, all TLV definitions are always aligned on an octet boundary and hence no padding is required. An MIS protocol payload carries a Source MISF Identifier TLV and a Destination MISF Identifier TLV followed by MIS Service Specific TLVs.

Figure 27 shows the components of the MIS protocol frame.



**Figure 27—MIS protocol general frame format**

The MIS protocol header (see Figure 28) carries the essential information that is present in every frame and is used for parsing and analyzing the MIS protocol frame.

|  |  |  |  |
| --- | --- | --- | --- |
| SID (4) | Opcode  (2) | AID (10) | |
|  | | | |
| Octet 1 | | | | | | | Octet 2 | | Octet 3 | | | Octet 4 |
|  | | | | | | |  | |  | | |  |
| Ver  (4) | | | Ack Reg (1) | Ack Reg (1) | UIR (1) | M (1) | FN (7) | Rsvd1 (1) | MIS Header ID (16) | | | |
| P (1) | S (1) | Rsvd2 (2) | Transaction ID (12) | | | | | | Variable Payload Length  (16) | | | |

**Figure 28—MIS protocol header format**

Table 23 shows the description of the header fields.

**Table 23—Description of MIS protocol header fields**

|  |  |  |
| --- | --- | --- |
| **Field name** | **Size (bits)** | **Description** |
| Version | 4 | This field is used to specify the version of MIS protocol used.  0: Not to be used 1: First version 2–15: (*Reserved*)  The version number will be incremented only when a fundamental incompatibility exists between a new revision and the prior edition of the standard. An MIS node that receives an MIS message with a higher version number than it supports will discard the frame with­out indication to the sending MIS node. |
| ACK-Req | 1 | This field is used for requesting an acknowledgement for the message. |
| ACK-Rsp | 1 | This field is used for responding to the request for an acknowledge­ment for the message. |

**Table 23—Description of MIS protocol header fields *(continued)***

|  |  |  |
| --- | --- | --- |
| **Field name** | **Size (bits)** | **Description** |
| Unauthenticated information | 1 | This field is used by the MIS Information Service to indicate if the |
| request (UIR) |  | protocol message is sent in pre-authentication/pre-association state so that the length of the response message can be limited. The UIR bit should be set to '1' by the originator when making an MIS infor­mation service request over a certain link in the un-associated/unau­thenticated or unregistered state. |
|  |  | In all other cases, this bit is set to '0'. |
| More fragment (M) | 1 | This field is used for indicating that the message is a fragment to be followed by another fragment. It is set to '0' for a message that is not fragmented and for the last fragment. The two 0 valued conditions are differentiated by the FN field. It is set to '1' for a fragment that is not the last one. |
| Fragment number (FN) | 7 | This field is used for representing the sequence number of a frag­ment. The fragment number starts from 0. The maximum fragment number is 127. This field is set to '0' for a message that is not fragmented. |
| Reserved1 | 1 | This field is intentionally kept reserved. When not used, this bit is set to '0'. |
| MIS message ID (MID) | 16 | Combination of the following 3 fields. |
| -- Service identifier (SID) | 4 | Identifies the different MIS services, possible values are as follows: |
|  |  | 1. Service Management |
|  |  | 1. Event Service |
|  |  | 1. Command Service |
|  |  | 1. Information Service |
| -- Operation code (Opcode) | 2 | Type of operation to be performed with respect to the SID, possible values are as follows: |
|  |  | 1. Request |
|  |  | 1. Response |
|  |  | 1. Indication |
| -- Action identifier (AID) | 10 | This indicates the action to be taken with regard to the SID (see |
|  |  | Table L. 1 for AID assignments). |
| Proactive authentication (P) | 1 | This field is used for indicating that the message is a proactive authentication message. |
| Security association (S) | 1 | This field is used for indicating that a security  association exists and the message is protected. |
| Reserved2 | 2 | This field is intentionally kept reserved. When not used, all the bits of this field are to be set to '0'. |
| Transaction ID | 12 | This field is used for matching Request and Response, as well as matching Request, Response and Indication to an ACK. |
| Variable payload length | 16 | Indicates the total length of the variable payload embedded in this |
|  |  | MIS protocol frame. The length of the MIS protocol header is NOT included. |

Protected MIS protocol frame format

In an MIS header the following two bits are used to indicate that an MIS PDU is protected and/or is used to carry a proactive authentication message.

a) P bit — Setting P bit to one indicates that the message carries a proactive authentication message.

b) S bit — Setting S bit to one indicates that an MIS security association exists and the service specific

TLVs are protected.

A protected MIS PDU is an MIS PDU that has an MIS header with S bit set to one indicating that the MIS service specific TLVs in this PDU are protected. Each security association is defined for a pair of MISF identifiers and is identified by a security association identifier (SAID). Therefore, for a protected MIS PDU, when a security association identifier is defined, the Source and Destination MISF identifier TLVs may not be present. In this case, an MIS header is followed by an SAID TLV, which is followed by a security TLV. Figure 28a shows a protected MIS protocol frame, where the source and destination MISF TLVs are optional.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MIS header  (S=1) | Source MISF Identifier TLV | Destination MISF Identifier TLV | SAID TLV | Security TLV |

**Figure 28a—Protected MIS frame format**

* + - * 1. MIS PDU protected by (D)TLS

MIS message can be protected using TLS [RFC5246] or DTLS [RFC4347].

The transport protocol for (D)TLS is the MIS protocol. When the MIS protocol transport is reliable, TLS is used. Otherwise, DTLS is used. The transport protocol entities to be associated with a TLS session are MISF peers and are identified by MISF identifiers. The TLS handshake takes place over the MIS protocol and as a result, an MIS SA that contains TLS master key and its child keys, TLS random values and the TLS cipher suite negotiated in the TLS handshake is established between the peers. The detailed description about the protocol interface of using (D)TLS is provided in 9.1.1.

The structure of an MISS PDU during a TLS handshake is shown in Figure 28b.

|  |  |  |  |
| --- | --- | --- | --- |
| MIS header | Source MISF Identifier TLV | Destination MISF Identifier TLV | Security TLV  (TLS record type = handshake/change cipher) |

**Figure 28b—MISS PDU during TLS handshake**

Once a (D)TLS handshake is completed, an MIS SA is established, which is determined by the ciphersuite negotiated in the (D)TLS handshake. The structure of protected MIS PDU, when an MIS SA exists, is shown in Figure 28c, where the unprotected MIS service specific TLVs are carried and protected as (D)TLS application data. An MIS header has the S bit set to one. The TLS session ID assigned through TLS hand- shake is contained in the SAID TLV. The TLS protection can be integrity protected, encrypted, or both. If it is integrity protected, then a message integrity code (MIC) is also included in the security TLV. In this stan- dard, the message integrity code is the same as the message authentication code, for which the acronym MAC is already used for media access control.

MIS Service

Specific TLVs

TLS Protection

MIS header

SAID TLV

Security TLV

(TLS record type = application data)

**Figure 28c—MISS PDU in Existence of MIS SA by TLS**

The MIS message protection procedure is specified in 9.3.

* + - * 1. MIS PDU protected through EAP-generated MIS SA

An MIS security association (SA) may be established through an MIS service access authentication. An MIS SA is established for a pair of MISFs. It includes a ciphersuite used for the protection. A security association identifier is assigned by the PoS as a result of successful EAP execution and communicated to the MN via a MIS\_Auth request message with a Status indicating Success. Figure 28d shows a protected MIS PDU. The protection procedure is specified in 9.3.1.

MIS Service

Specific TLVs

Protection through EAP-generated SA

MIS header

SAID TLV

Security TLV

Figure 28d—MIS PDU Protected by an EAP-generated MIS SA

* + - * 1. Protected MIS PDU upon transport address change

If the transport address of an MISF peer changes over the lifetime of a TLS session or the lifetime of an SA, the mapping between the sender's transport address and the MISF identifier shall be updated only if the MISF identifier is the same as that is currently bound to the security association identifier, and an MIS reg- istration request or response message may be contained in the Security TLV. The structure of a protected MIS PDU upon transport address change is shown in Figure 28e.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MIS header | Source MISF Identifier TLV | Destination MISF Identifier TLV | SAID TLV | Security TLV |

**Figure 28e—MIS PDU upon Transport Address Change**

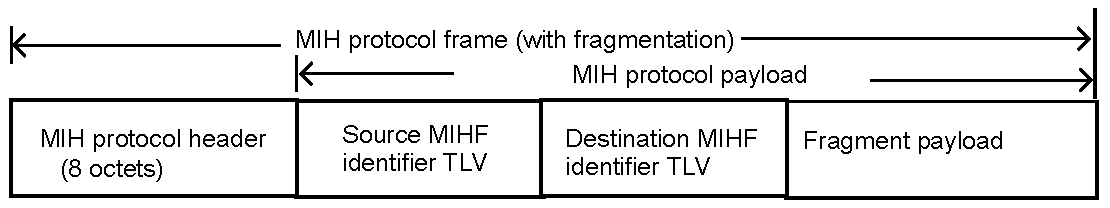
Fragmentation and reassembly

* + - * 1. General

The MIS fragmentation mechanism is defined using 'M' (More Fragment) and 'FN' (Fragment Number) fields of the MIS protocol header.

An MIS message is fragmented only when MIS message is sent natively over an L2 medium such as Ethernet. The message is fragmented when the message size exceeds aFragmentationThreshold. The size of each of the fragments is the same except the last one, which may be smaller. The maximum fragment size is defined as the maximum value of aFragmentationThreshold, which shall be equal to the Maximum Transmission Unit (MTU) of the link layer that is on the path between two MISF nodes, minus securityOverhead octets, which is the maximum expansion for each protected MIS PDU. When there is no MIS SA, securityOverhead is set to zero. The calculation of securityOverhead when there is an MIS SA is given in Annex K. When the MTU of the link layer between two MISF nodes is known, the maximum fragment size is set to the MTU (in octets) minus securityOverhead octets. The method of determining such an MTU is outside the scope of this standard. When the MTU of the link layer between two MISF nodes is unknown, the maximum fragment size is set to the minimum MTU of 1500 octets minus securityOverhead octets. When MIS message is sent using an L3 or higher layer transport, L3 takes care of any fragmentation issue, and the MIS protocol does not handle fragmentation in such cases.

Figure 29 shows the components of the fragmented MIS protocol frame. The MIS protocol payload carries a Source MISF Identifier TLV and a Destination MISF Identifier TLV followed by a fragment payload. Based on the fragment size, the fragment payload may not be aligned on a TLV boundary, i.e., TLVs other than the source MISF identifier and destination MISF identifier TLVs may not be complete within the fragment payload. The fragment size may be smaller than the maximum fragment size and shall be larger than the value that can generate more than 128 fragments.



**Figure 29—Fragmented MIS protocol frame format**

When an MIS PDU is protected, the protection is applied to the fragment payload as shown in Figure 29a.

Fragment payload

Protection

((D)TLS or EAP-generated SA)

MIS header

SAID TLV

Security TLV (Protected fragment payload)

**Figure 29a—Protected fragmented MIS protocol frame format**

* + - * 1. Fragmentation

When an MIS message is fragmented, the fragmentation is performed within 'Transmit()' procedure in the MIS transaction protocol state machines. The MIS protocol header, the source MISF identifier TLV and destination MISF identifier TLV of the original message are copied to each fragment. When an MIS SA exists, the S bit in the header is set to one and an SAID TLV is included in each fragment. In this case, the source MISF identifier TLV and destination MISF identifier TLV of the original message are optional. However the 'variable payload length', 'more fragment', and 'fragment number' fields are updated accordingly for each fragment.

Variable payload length of each fragment indicates the number of octets in the MIS protocol payload of that fragment.

'More fragment' and 'fragment number' fields of each fragment are set according to the description in Table 23.

When data are to be transmitted, the number of octets in the fragment shall be determined by the fragment size and the number of octets in the multi-fragment message that have yet to be assigned to a fragment at the instant the fragment is constructed for the first time. Once a fragment is transmitted for the first time, its frame body content and length shall be fixed until it is successfully delivered to the destination MISF.

No retransmission by the MIS protocol (defined in 8.2) is performed for any single fragment of a multi- fragment message.

* + - * 1. Reassembly

The destination MISF reassembles the received fragments into an original message. Reassembly is performed outside the MIS transaction state machines. 'MsgIn' and 'MsgInAvail' variables are set only after successful reassembly. An MISF shall be capable of receiving fragments of arbitrary length.

The following fields are used for reassembling fragments:

* S bit
* MIS message ID
* Transaction ID
* Source MISF identifier TLV
* Destination MISF identifier TLV
* SAID TLV (when Source and Destination MISF identifiers are not present)
* More fragment
* Fragment number

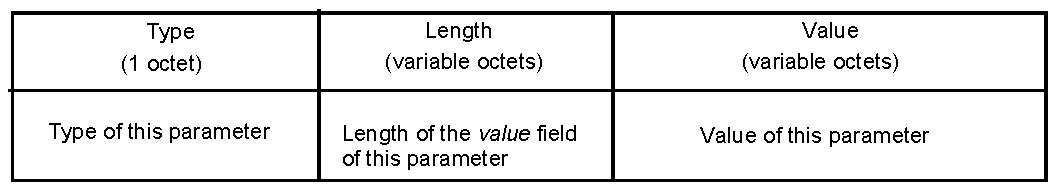
When any fragment of a multi-fragment message has arrived first, the destination MISF starts a timer referred to as ReassemblyTimer. If this ReassemblyTimer expires before all fragments have been received, the destination MISF discards those fragments that it has received. A duplicate fragment is discarded.

When S bit is set to one, the fragment is protected. The protected fragment is verified for its integrity at the receiving end. If encryption is applied, it is decrypted to obtain the plaintext fragment. The security association identifier maps the fragment to a pair of source and destination MISF identifiers that are required for reassembly. The reassembly is performed after all the fragments are verified and decrypted.

An example of an original MIS message and fragmented MIS messages is shown in Annex K.

* 1. Message parameter TLV encoding

The following general TLV encoding (shown in Figure 30) shall be used for all parameters in an MIS protocol message.

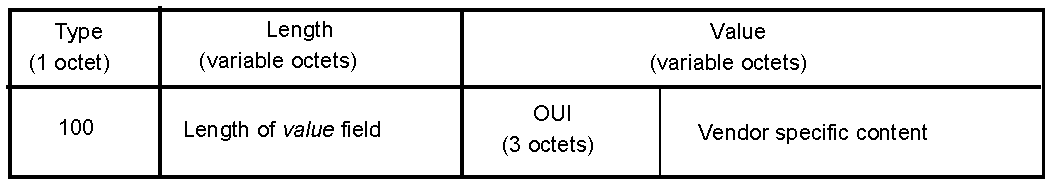


**Figure 30—Message parameter TLV encoding**

Specifically, the *Type* field is one octet13, and the *Length* shall be encoded with the rules described in 6.5.6.2.

Moreover, TLV *Type* values shall be unique within the MIS protocol. The TLV encoding starts at 1 and any subsequent values are assigned in ascending order (see Annex L, Table L.2).

The TLV encoding of the vendor specific TLV (type = 100) is shown in Figure 31.



**Figure 31—The TLV encoding for the vendor specific TLV (Type = 100)**

13The TLV *Type* field length is different than the Information Element *Type* length, which is four octets.

* 1. MIS protocol messages

The following subclauses specify different MIS protocol messages in TLV form. The shaded areas represent the MIS protocol header, while the unshaded areas represent the MIS protocol payload. The payload consists of a set of identifiers in TLV form.

The TLV Type assignment for each TLV can be found in Annex L, Table L.2.

TLV type values ranging from 101 to 255 are reserved for experimental TLVs. These values are used by different implementations to evaluate the option of using TLVs not defined by the specification.

When a TLV type value is in the range of experimental TLVs and the data type of the TLV value is unknown or the TLV value is not in the range of valid values, the TLV should be ignored and the rest of the message should be processed. Also, experimental TLVs can be ignored, based on the MISF information that is communicating with another MISF with different experimental TLVs implementation.

All MIS messages carry a source MISF ID followed by a destination MISF ID as the first two TLVs of the MIS protocol payload part of the message. Multicast MISF ID can be used in MIS\_Capability\_Discover request and response messages as its destination MISF ID.

All “Optional” fields are optionally sent but the receiver shall properly operate on them if present, i.e., these fields are mandatory in the implementation, but optional in their use.

On receipt of an MIS request message the MISF shall respond with a corresponding response message.

Any message received that has an invalid MIS header, or does not contain the source/destination MISF IDs, or has an unrecognizable or invalid MIS Message ID shall be discarded without sending any indication to the source MIS node. Any undefined or unrecognizable TLVs in a received message shall be ignored by the receiver.

MIS messages for service management

* + - * 1. MIS\_Capability\_Discover request

The corresponding MIS primitive of this message is defined in 7.4.1.1.

If a requesting MISF entity knows the destination MISF entity’s MISF ID, the requesting MISF entity fills its destination MISF ID and sends this message to the peer MISF over the data plane, either L2 or L3.

If a requesting MISF entity does not know the destination MISF entity’s MISF ID, the requesting MISF entity may fill its destination MISF ID with a multicast MISF ID to send this capability discover message.

|  |
| --- |
| **11IH Header Fields (SID=1, Opcode=1, AID=1)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkAddressList (optional) (Link address list TLV) |
| SupportedMisEventList (optional) (MIS event list TLV) |
| SupportedMisCommandList (optional) (MIS command list TLV) |
| SupportedISQueryTypeList (optional) (MIIS query type list TLV) |
| SupportedTransportList (optional) (Transport option list TLV) |
| MBBHandoverSupport (optional) (MBB handover support TLV) |
| SupportedSecurityCapList (optional)  (MIS Service Authentication Method list TLV) |

* + - * 1. MIS\_Capabil ity\_Discover response

The corresponding MIS primitive of this message is defined in 7.4.1.3. This message is sent in response to an MIS\_Capability\_Discover request message that was destined to a single or multicast MISF ID.

|  |
| --- |
| **11IH Header Fields (SID=1, Opcode=2, AID=1)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| Status (Status TLV) |
| Link Address List (optional) (Link address list TLV) |
| SupportedMisEventList (optional) (MIS event list TLV) |
| SupportedMisCommandList (optional) (MIS command list TLV) |
| SupportedISQueryTypeList (optional) (MIIS query type list TLV) |
| SupportedTransportList (optional) (Transport option list TLV) |
| MBBHandoverSupport (optional) (MBB handover support TLV) |
| SupportedSecurityCapList (optional)  (MIS Service Authentication Method list TLV) |

* + - * 1. MIS\_Register request

The corresponding MIS primitive of this message is defined in 7.4.2.1.

This message is transmitted to the remote MISF to perform a registration or re-registration.

|  |
| --- |
| **11IH Header Fields (SID=1, Opcode=1, AID=2)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkIdentifierList (Link identifier list TLV) |
| RequestCode (Register request code TLV) |

* + - * 1. MIS\_Register response

The corresponding MIS primitive of this message is defined in 7.4.2.3. This message is sent in response to a registration or re-registration request.

|  |
| --- |
| **11IH Header Fields (SID=1, Opcode=2, AID=2)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| Status (Status TLV) |
| ValidTimeInterval (not included if Status does not indicate “Success”) (Valid time interval TLV) |

* + - * 1. MIS\_DeRegister request

The corresponding MIS primitive of this message is defined in 7.4.3.1.

This message is transmitted to the remote MISF to request a de-registration. There is no parameter for this message.

**11IH Header Fields (SID=1, Opcode=1, AID=3)**

**Source Identifier =** sending MISF ID  
(Source MISF ID TLV)

**Destination Identifier =** receiving MISF ID  
(Destination MISF ID TLV)

* + - * 1. MIS\_DeRegister response

The corresponding MIS primitive of this message is defined in 7.4.3.3.

This message is sent in response to a de-registration request.

|  |
| --- |
| **11IH Header Fields (SID=1, Opcode=2, AID=3)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| Status (Status TLV) |

* + - * 1. MIS\_Event\_Subscribe request

The corresponding MIS primitive of this message is defined in 7.4.4.1.

This message is sent by a remote MISF (the subscriber) to subscribe to one or more event types from a particular event origination point.

|  |
| --- |
| **11IH Header Fields (SID=1, Opcode=1, AID=4)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkIdentifier (Link identifier TLV) |
| RequestedMisEventList (MIS event list TLV) |
| EventConfigurationInfoList (optional) (Event configuration info list TLV) |

* + - * 1. MIS\_Event\_Subscribe response

The corresponding MIS primitive of this message is defined in 7.4.4.2.

The response indicates which of the event types were successfully subscribed.

|  |
| --- |
| **11IH Header Fields (SID=1, Opcode=2, AID=4)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| Status (Status TLV) |
| LinkIdentifier (Link identifier TLV) |
| ResponseMisEventList (not included if Status does not indicate “Success”) (MIS event list TLV) |

* + - * 1. MIS\_Event\_Unsubscribe request

The corresponding MIS primitive of this message is defined in 7.4.5.1.

This message is sent by a remote MISF (the subscriber) to unsubscribe from a set of link-layer events.

|  |
| --- |
| **MIS Header Fields (SID=1, Opcode=1, AID=5)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkIdentifier (Link identifier TLV) |
| RequestedMisEventList (MIS event list TLV) |

* + - * 1. MIS\_Event\_Unsubscribe response

The corresponding MIS primitive of this message is defined in 7.4.5.2.

The response indicates which of the event types were successfully unsubscribed.

|  |
| --- |
| **MIS Header Fields (SID=1, Opcode=2, AID=5)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| Status (Status TLV) |
| LinkIdentifier (Link identifier TLV) |
| ResponseMisEventList (not included if Status does not indicate “Success”) (MIS event list TLV) |

**8.6.1.11 MIS\_Auth indication**

This is used for an MISF to perform (D)TLS exchange with another MISF to establish or terminate a (D)TLS-generated MIS SA. It is also used to initiate an MIS service access authentication through EAP or ERP. In the former case, an AuthenticationContent shall be included to carry a TLS record of type handshake, change ciphersuite or alert message. In the latter case, this message is used in two different situations: a) when EAP execution is initiated by the MN; b) when ERP execution is initiated by the PoS. Only in case b), AuthenticationContent shall be included to carry an ERP message (ERP-Initiate/Re-auth- Start). This message shall not be used when EAP execution is initiated by a PoS or when ERP execution is initiated by an MN; an MIS\_Auth request message shall be used instead.

|  |
| --- |
| **MIS Header Fields (SID=1, Opcode=3, AID=6)** |
| **Source Identifier** = sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier** = receiving MISF ID (Destination MISF ID TLV) |
| AuthenticationContent (optional) (Authentication TLV) |

**8.6.1.12 MIS\_Auth request**

This message is used for an MISF in either an MN or a PoS to send EAP or ERP messages in an MIS ser- vice authentication.

|  |
| --- |
| **MIS Header Fields (SID=1, Opcode=1, AID=6)** |
| **Source Identifier** = sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier** = receiving MISF ID (Destination MISF ID TLV) |
| Security association ID (optional) (SAID TLV) |
| Nonce (optional) (Nonce TLV) |
| AuthenticationContent (optional) (Authentication TLV) |

|  |
| --- |
| KeyLifeTime (optional) (Lifetime TLV) |
| Status (optional) (STATUS TLV) |
| CipherSuite(optional) (Ciphersuite TLV) |
| AUTH (optional) (AUTH TLV) |

**8.6.1.13 MIS\_Auth response**

This message is used for an MISF in either an MN or a PoS to send EAP or ERP messages in an MIS ser- vice authentication.

|  |
| --- |
| **MIS Header Fields (SID=1, Opcode=2, AID=6)** |
| **Source Identifier** = sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier** = receiving MISF ID (Destination MISF ID TLV) |
| Nonce (optional) (Nonce TLV) |
| AuthenticationContent (optional) (Authentication TLV) |
| KeyLifeTime (optional) (Lifetime TLV) |
| Status (optional) (STATUS TLV) |
| CipherSuite(optional) (Ciphersuite TLV) |
| AUTH (optional) (AUTH TLV) |

**8.6.1.14 MIS\_Termination\_Auth request**

This message is used for an MISF in a PoS to terminate an MIS SA.

**MIS Header Fields (SID=1, Opcode=1, AID=7)**

**Source Identifier** = sending MISF ID (Source MISF ID TLV)

**Destination Identifier** = receiving MISF ID (Destination MISF ID TLV)

**8.6.1.15 MIS\_Termination\_Auth response**

This message is used for an MISF in an MN to terminate an MIS SA.

**MIS Header Fields (SID=1, Opcode=2, AID=7)**

**Source Identifier** = sending MISF ID (Source MISF ID TLV)

**Destination Identifier** = receiving MISF ID (Destination MISF ID TLV)

**8.6.1.16 MIS\_Push\_key request**

This message is used for an MISF to communicate to another MISF to push a media specific master session key or media specific master session keys to a specific PoA or PoAs. The corresponding primitive is defined in 7.4.27.1.

|  |
| --- |
| **MIS Header Fields (SID=1, Opcode=1, AID=8)** |
| **Source Identifier** = sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier** = receiving MISF ID (Destination MISF ID TLV) |
| LinkTupleIdentifierList  (Link Tuple Identifier List TLV) |

**8.6.1.17 MIS\_Push\_key response**

This message is used for an MISF to communicate to another MISF that a media specific master session key or media specific master session keys are installed in a specific PoA or PoAs. The corresponding primitive is defined in 7.4.27.3.

|  |
| --- |
| **MIS Header Fields (SID=1, Opcode=2, AID=8)** |
| **Source Identifier** = sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier** = receiving MISF ID (Destination MISF ID TLV) |
| LinkTupleIdentifierList  (Link Tuple Identifier List TLV) |
| Status (optional) (STATUS TLV) |

**8.6.1.18 MIS\_LL\_Auth request**

This message is used for an MISF to carry link layer frames to conduct an authentication. The correspond- ing primitive is defined in 7.4.28.1.

|  |
| --- |
| **MIS Header Fields (SID=1, Opcode=1, AID=9)** |
| **Source Identifier** = sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier** = receiving MISF ID (Destination MISF ID TLV) |
| LinkIdentifier  (Link Identifier TLV) |
| LLInformation  (Link Layer Information TLV) |

**8.6.1.19 MIS\_LL\_Auth response**

This message is used for an MISF to carry link layer frames to conduct an authentication. The correspond- ing primitive is defined in 7.4.28.3.

|  |
| --- |
| **MIS Header Fields (SID=1, Opcode=2, AID=9)** |
| **Source Identifier** = sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier** = receiving MISF ID (Destination MISF ID TLV) |
| LinkIdentifier  (Link Identifier TLV) |
| LLInformation  (Link Layer Information TLV) |
| Status  (Status TLV) |

MIS messages for event service

* + - * 1. MIS\_Link\_Detected indication

The corresponding MIS primitive of this message is defined in 7.4.6.

This message is transmitted to the remote MISF when a new link has been detected.

|  |
| --- |
| **11IH Header Fields (SID=2, Opcode=3, AID=1)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkDetectedInfoList (Link detected info list TLV) |

* + - * 1. MIS\_Link\_Up indication

The corresponding MIS primitive of this message is defined in 7.4.7.

This notification is delivered from an MISF, when present in the PoA, to an MISF in the network when a layer 2 connection is successfully established with an MN.

|  |
| --- |
| **11IH Header Fields (SID=2, Opcode=3, AID=2)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkIdentifier (Link identifier TLV) |
| OldAccessRouter (optional) (Old access router TLV) |
| NewAccessRouter (optional) (New access router TLV) |
| IPRenewalFlag (optional) (IP renewal flag TLV) |
| MobilityManagementSupport (optional) (Mobility management support TLV) |

* + - * 1. MIS\_Link\_Down indication

The corresponding MIS primitive of this message is defined in 7.4.8.

This notification is delivered from an MISF, when present in the PoA, to an MISF in the network when a layer 2 connection with an MN is disconnected due to a certain reason.

|  |
| --- |
| **MIS Header Fields (SID=2, Opcode=3, AID=3)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkIdentifier (Link identifier TLV) |
| OldAccessRouter (optional) (Old access router TLV) |
| ReasonCode (Link down reason code TLV) |

* + - * 1. MIS\_Link\_Parameters\_Report indication

The corresponding MIS primitive of this message is defined in 7.4.9.

This message indicates changes in link conditions that have crossed pre-configured threshold levels. A pre- configured threshold level is set by the MIS\_Link\_Configure\_Thresholds request message.

|  |
| --- |
| **MIS Header Fields (SID=2, Opcode=3, AID=5)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkIdentifier (Link identifier TLV) |
| LinkParameterReportList (Link parameter report list TLV) |

* + - * 1. MIS\_Link\_Going\_Down indication

The corresponding MIS primitive of this message is defined in 7.4.10.

This message is transmitted to the remote MISF when a layer 2 connectivity is expected (predicted) to go down within a certain time interval.

|  |
| --- |
| **MIS Header Fields (SID=2, Opcode=3, AID=6)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkIdentifier (Link identifier TLV) |
| TimeInterval (Time interval TLV) |
| LinkGoingDownReason (Link going down reason TLV) |

* + - * 1. MIS\_Link\_Handover\_Imminent indication (ISSUE: mark these sections as excluded)
        2. MIS\_Link\_Handover\_Complete indication

MIS messages for command service

* + - * 1. MIS\_Link\_Get\_Parameters request

The corresponding MIS primitive of this message is defined in 7.4.14.2.

This message is used to discover the status of currently available links.

|  |
| --- |
| **11IH Header Fields (SID=3, Opcode=1, AID=1)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| DeviceStatesRequest (optional) (Device states request TLV) |
| LinkIdentifierList (Link identifier list TLV) |
| GetStatusRequestSet (Get status request set TLV) |

* + - * 1. MIS\_Link\_Get\_Parameters response

The corresponding MIS primitive of this message is defined in 7.4.14.3.

This message is used by an MISF to report the status of currently available links.

|  |
| --- |
| **11IH Header Fields (SID=3, Opcode=2, AID=1)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| Status (Status TLV) |
| DeviceStatesResponseList (optional) (not included if Status does not indicate “Success”) (Device states response list TLV) |
| GetStatusResponseList (not included if Status does not indicate “Success”) (Get status response list TLV) |

* + - * 1. MIS\_Link\_Configure\_Thresholds request

The corresponding MIS primitive of this message is defined in 7.4.15.2. This message is used to configure thresholds of the lower layer link.

|  |
| --- |
| **11IH Header Fields (SID=3, Opcode=1, AID=2)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkIdentifier (Link identifier TLV) |
| ConfigureRequestList (Configure request list TLV) |

* + - * 1. MIS\_Link\_Configure\_Thresholds response

The corresponding MIS primitive of this message is defined in 7.4.15.3.

This message returns the status of a thresholds configuration request. The MISF generating this message generates MIS\_Link\_Parameters\_Report indication message when the configured threshold is crossed.

|  |
| --- |
| **11IH Header Fields (SID=3, Opcode=2, AID=2)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| Status (Status TLV) |
| LinkIdentifier (Link identifier TLV) |
| ConfigureResponseList (not included if Status does not indicate “Success”) (Configure response list TLV) |

* + - * 1. MIS\_Link\_Actions request

The corresponding MIS primitive of this message is defined in 7.4.16.1. This message is used to control the behavior of a set of lower layer links.

|  |
| --- |
| **11IH Header Fields (SID=3, Opcode=1, AID=3)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| LinkActionsList (Link actions list TLV) |

* + - * 1. MIS\_Link\_Actions response

The corresponding MIS primitive of this message is defined in 7.4.16.2. This message returns the result of an MIS\_Link\_Actions request.

|  |
| --- |
| **11IH Header Fields (SID=3, Opcode=2, AID=3)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| Status (Status TLV) |
| LinkActionsResultList (not included if Status does not indicate “Success”) (Link actions result list TLV) |

* + - * 1. MIS\_Net\_HO\_Candidate\_Query request (to be excluded)
        2. MIS\_Net\_HO\_Candidate\_Query response (to be excluded)
        3. MIS\_M N\_HO\_Candidate\_Query request (to be excluded)
        4. MIS\_MN\_HO\_Candidate\_Query response (to be excluded)
        5. MIS\_N2N\_HO\_Query\_Resources request (to be excluded)
        6. MI H\_N2N\_HO\_Query\_Resources response (to be excluded)
        7. MIS\_MN\_HO\_Commit request (to be excluded)
        8. MIS\_MN\_HO\_Commit response (to be excluded)
        9. MIS\_Net\_HO\_Commit request (to be excluded)
        10. MIS\_Net\_HO\_Commit response (to be excluded)
        11. MIS\_N2N\_HO\_Commit request (to be excluded)
        12. MIS\_N2N\_HO\_Commit response (to be excluded)
        13. MIS\_MN\_HO\_Complete request (to be excluded)
        14. MIS\_MN\_HO\_Complete response (to be excluded)
        15. MIS\_N2N\_HO\_Complete request (to be excluded)
        16. MIS\_N2N\_HO\_Complete response (to be excluded)

MIS messages for information service

MIS Information service uses only the following messages—MIS\_Get\_Information request, MIS\_Get\_Information response, and MIS\_Push\_Information. Due to the need to support different query types and the need for flexibility to customize the query and response, the parameters and their usage in these messages are substantially different from other MIS message parameters, and are therefore separately defined in the following subclauses.

* + - * 1. MIS\_Get\_Information request

The corresponding MIS primitive of this message is defined in 7.4.25.1.

This message is used by an MISF to retrieve a set of Information Elements provided by the information service. A single MIS\_Get\_Information request message carries only one query list. However, there can be multiple queries in that list in the order of the most preferred query first.

|  |
| --- |
| **MIS Header Fields (SID=4, Opcode=1, AID=1)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| InfoQueryBinaryDataList (optional) (Info query binary data list TLV) |
| InfoQueryRDFDataList (optional) (Info query RDF data list TLV) |
| InfoQueryRDFSchemaURL (optional) (Info query RDF schema URL TLV) |
| InfoQueryRDFSchemaList (optional) (Info query RDF schema list TLV) |
| MaxResponseSize (optional) (Max response size TLV) |
| QuerierNetworkType (optional) (Network type TLV) |
| UnauthenticatedInformationRequest (Unauthenticated information request TLV) |

* + - * 1. MI H\_Get\_Information response

The corresponding MIS primitive of this message is defined in 7.4.25.3.

This is used as a response to the MIS\_Get\_Information request message. The total response message size shall not exceed the value indicated in the Max Response Size TLV of corresponding MIS\_Get\_Information request message. The order of the query response shall be in the same order as the query requests.

|  |
| --- |
| **11IH Header Fields (SID=4, Opcode=2, AID=1)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| Status (Status TLV) |
| InfoResponseBinaryDataList (optional) (Info response binary data list TLV) |
| InfoResponseRDFDataList (optional) (Info response RDF data list TLV) |
| InfoResponseRDFSchemaURLList (optional) (Info response RDF schema URL list TLV) |
| InfoResponseRDFSchemaList (optional) (Info response RDF schema list TLV) |

* + - * 1. MIS\_Push\_Information indication

The corresponding MIS primitive of this message is defined in 7.4.26.1.

This is an indication to push operator policies or other network information to the MN.

|  |
| --- |
| **11IH Header Fields (SID=4, Opcode=3, AID=2)** |
| **Source Identifier =** sending MISF ID (Source MISF ID TLV) |
| **Destination Identifier =** receiving MISF ID (Destination MISF ID TLV) |
| InfoResponseBinaryDataList (optional) (Info response binary data list TLV) |
| InfoResponseRDFDataList (optional) (Info response RDF data list TLV) |
| InfoResponseRDFSchemaURLList (optional) (Info response RDF schema URL list TLV) |
| InfoResponseRDFSchemaList (optional) (Info response RDF schema list TLV) |

Xxx

**9. MIS protocol protection**

This clause specifies options and mechanisms to protect remote messages in the media independent services protocol. The remote messages in the MIS protocol can be protected through the transport protocols at layer 2 or layer 3. The protection through the transport protocols are discussed in Annex O. This clause specifies the mechanisms to protect MIS PDUs at the MIS layer. These mechanisms apply protection to MIS PDUs without depending on transport protocols. They are called MIS specific protection mechanisms. To apply MIS specific protection mechanisms, a MN and a point of service (PoS) need to negotiate protection mechanisms and to establish cryptographic keys. MIS message protection shall be accomplished in either of two ways. The first is to use TLS or DTLS and the other is to use EAP or ERP as an MIS service access authentication to establish MIS security associations (SAs). If MIS service access authentication is needed and an authentication server is available, then EAP based authentication and key establishment may be used for establishing an MIS SA. In situations where MIS service access authentication is not required and TLS credentials are available or where MIS service access authentication is required and TLS credentials for access authentication are available at a PoS, then (D)TLS may be used for establishing an MIS SA.

**9.1 Protection established through MIS (D)TLS**

In this option, a MN, the client, and a PoS, the server, execute a TLS, specified in IETF RFC 5246, or DTLS, specified in IETF RFC 4347, to establish MIS protection. When the MIS protocol transport is reliable, TLS is used. Otherwise, DTLS is used. In the rest of this standard, (D)TLS is used to denote TLS or DTLS. In a (D)TLS handshake, the mutual authentication is executed through either a pre-shared key or a public key certified by a trusted third party such as a certificate authority. It should be noted that all certificates are required to be validated. The TLS certificate used by the PoS is expected to be provided to the MN in a secure manner, e.g., during provisioning process. In this option, the authentication may or may not be related to access control. It can be an access authentication for MIS service if a PoS holds service credentials for the MNs.

After the handshake, a (D)TLS session is established. In this case, the TLS master key and the keys derived from the master key, all the TLS parameters, and TLS ciphersuite negotiated in the TLS handshake form an MIS SA. The (D)TLS security association identifier is carried in each message in the SAID TLV.

In a (D)TLS session, an MIS message is first protected as application data. Then the (D)TLS record is transported by MIS protocol by security TLV.

For a (D)TLS-generated MIS SA, it can be terminated through (D)TLS session termination using an

MIS\_Auth indication message.

**9.2 Key establishment through an MIS service access authentication**

If MIS service is subscription based and provided by a service provider, then an MIS service access authentication may be needed to authorize the service to a MN. In this case, a PoS may obtain a master session key through service access authentication and an MIS security associations can be established through the master session key between the MN and the PoS.

**9.2.1 MIS service access authentication**

In this standard, it is assumed that EAP [IETF RFC3748] or EAP Re-authentication (ERP) IETF RFC5296 is used as the authentication protocol with an MN as the peer and a PoS as the authenticator. An EAP server may be used as a backend server.

For the interface between an MN and a PoS, the MIS protocol is acting as an EAP lower layer. That is, at the MN, an EAP message is generated at the MISF. When it reaches the PoS, the MISF in the PoS will process it. For an EAP message from the PoS to the MN, it will also be generated by the MISF in the PoS. At the MN, the EAP message is passed to the MISF to process. The protocol stack is illustrated in Figure 30, where it is assumed that an EAP server is employed. After a successful authentication, a master session key (MSK) is exported to the lower layer, that is, MIS layer. An MSK is used to further derive MIS message protection keys.

EAP Peer / MN EAP Authenticator / PoS

EAP Server/ AS

MISF MISF

EAP Method Layer

EAP Peer layer

EAP Method Layer

EAP Authenticator Layer

EAP Method

Layer

EAP Authenticator

Layer

EAP Layer EAP Layer

EAP Layer

MSK

MIS PDU

MSK MSK

MSK

MSK

EAP lower-layer

(MIS Protocol)

EAP Lower-layer

(MIS Protocol)

AAA / IP

AAA / IP

**Figure 30—Protocol Stack of Service Access Authentication (with an EAP Server)**

The authentication is divided into the following phases:

a) *Capability Discovery Phase*. In this phase, both the MN and the PoS exchange unprotected MIS

messages for an MN to discover the services that a PoS provides.

b) *MIS Service Access Authentication Phase*. Before starting the MIS access authentication, the MN and the PoS perform a negotiation in order to agree on a ciphersuite and other useful parameters to be used in the authentication and MIS message protection. The negotiation may be initiated either by the MN or by the PoS. Once the negotiation is completed, the MN (acting as the EAP peer) authenticates against the PoS (acting as an EAP authenticator). To achieve this, EAP is transported by MIS protocol between the MN and the PoS. In order to carry out the authentication the PoS may use a backend authentication server (acting as an EAP server) to verify the MN’s credentials. In this standard, it is assumed that the EAP methods employed can export keying material (i.e., MSK). Thus, after performing the authentication, keying material (i.e., MSK) will be shared between the MN and the PoS. Specifically, the keying material is exported to MN’s and PoS's lower layer (MIS layer) and used to protect the rest of the communication. The message protection mechanisms are specified in 9.3. The protected message format is specified in 8.4. In order to preserve the security of

the exported keying material, the exported MSK is used as a root key to derive session keys which are used to protect the MIS PDUs. The key hierarchy is described in 9.2.2. Note that the authentica- tion procedure could be based on an EAP re-authentication (ERP) in order to perform a fast authen- tication. In this case, an rMSK is used as the root key to derive the key hierarchy.

c) *Service Access Phase*. At this point, the MN is authenticated and authorized to use the MIS services, agreed and provided by the PoS. The MIS protocol is protected by using the keying material obtained in the MIS Service Access Authentication Phase. This phase is related to 9.2.2 for key derivation and 9.3 for protecting MIS protocol.

d) *Termination phase*. When the MN or the PoS desires to terminate the security association before the security association lifetime expires, either the MN or the PoS can request to terminate.

Figure 31 and Figure 32 illustrate the EAP execution when it is initiated by the MN and when it is initiated by the PoS respectively. In both figures, only the protocol interface between an EAP peer and an EAP authenticator is described. The interface with EAP server is not illustrated. MIS service access authentica- tion messages are defined in 8.6.1.11, 8.6.1.12, and 8.6.1.13. Termination messages are defined in 8.6.1.14 and 8.6.1.15.

Similarly, Figure 33 illustrates an MN initiated ERP execution. Figure 34 and Figure 35 show a PoS initi- ated ERP execution, where the ERP may be initiated by sending an EAP Request/Identity as shown in Figure 34 or by sending an ERP-Initiate/Re-auth-Start as shown in Figure 35.

EAP Peer

MN

EAP Authenticator

PoS

Capability Discovery

Phase

MIS Capability Discovery Request

MIS Capability Discovery Response

MIS Auth indication

MIS Auth Request (EAP, Ciphersuite)

MIS Service Access Authentication Phase

MIS Auth Response (EAP, Ciphersuite)

...

MIS Auth Request (EAP-success, SAID, AUTH)

MIS Auth Response (AUTH)

Service Access

Phase

Termination

Phase

...

MIS Termination Request

MIS Termination Response

**Figure 31—Main Stages with MN Initiated EAP Execution**

EAP Peer

MN

EAP Authenticator

PoS

Capability Discovery

Phase

MIS Capability Discovery Request

MIS Capability Discovery Response

Trigger

MIS Auth Request (EAP, Ciphersuite)

MIS Service Access Authentication Phase

MIS Auth Response (EAP, Ciphersuite)

...

MIS Auth Request (EAP-success, SAID, AUTH)

MIS Auth Response (AUTH)

Service Access

Phase

Termination

Phase

...

MIS Termination Request

MIS Termination Response

**Figure 32—Main Stages with PoS Initiated EAP Execution**

EAP Peer

MN

EAP Authenticator

PoS

Capability Discovery

Phase

MIS Capability Discovery Request

MIS Capability Discovery Response

MIS Auth Request (EAP-Initiate/Re-Auth, Ciphersuite)

MIS Service Access Authentication Phase

MIS Auth Response (EAP-Finish/Re-Auth, Ciphersuite)

MIS Auth Request (AUTH)

MIS Auth Response (AUTH)

Service Access

Phase

Termination

Phase

...

MIS Termination Request

MIS Termination Response

**Figure 33—Main Stages with MN Initiated ERP Execution**

EAP Peer

MN

EAP Authenticator

PoS

Capability Discovery

Phase

MIS Capability Discovery Request

MIS Capability Discovery Response

Trigger

MIS Auth indication (EAP-initiiate/Re-Auth Start)

MIS Service Access Authentication Phase

MIS Auth Request (EAP-Initiate, Ciphersuite)

MIS Auth Response (EAP-Finish, Ciphersuite)

MIS Auth Request (AUTH)

MIS Auth Response (AUTH)

Service Access

Phase

Termination

Phase

...

MIS Termination Request

MIS Termination Response

**Figure 34—Main Stages with PoS Initiated ERP Execution (1)**

EAP Peer

MN

EAP Authenticator

PoS

Capability Discovery

Phase

MIS Capability Discovery Request

MIS Capability Discovery Response

MIS Auth request (EAP-Request/Identity) MIS Auth response

MIS Service Access Authentication Phase

MIS Auth request (EAP-Initiate/Re-Auth, Ciphersuite)

MIS\_Auth response(EAP\_Finish/Re-Auth, Ciphersuite)

MIS Auth Request (AUTH)

MIS Auth Response (AUTH)

Service Access

Phase

Termination

Phase

...

MIS Termination Request

MIS Termination Response

**Figure 35—Main Stages with PoS Initiated ERP Execution (2)**

**9.2.2 Key derivation and key hierarchy**

Upon a successful MIS service access authentication, the authenticator, PoS, obtains a master session key (MSK) or a re-authentication master session key (rMSK). The keys derived from the MSK or rMSK include a 128 bit authentication key (MIAK) used to generate a value AUTH and the session keys determined by the ciphersuite agreed upon between an MN and a PoS. The session keys used for MIS message protection con- sist of an encryption key (MIEK) only, an integrity key (MIIK) only, or both an encryption key (MIEK) and an integrity key (MIIK). The length, *L,* of the derived keying material, called media independent session key (MISK) are specified in 9.2.3.

For the key derivation, the following notations and parameters are used.

— *K* - key derivation key. It is truncated from a master session key (MSK) or re-authentication MSK (rMSK). The length of *K* is determined by the pseudorandom function (PRF) used for key derivation. If HMAC-SHA-1 or HMAC-SHA-256 is used as a PRF, then the full MSK or rMSK is used as key

derivation key, *K*. If CMAC-AES is used as a PRF, then the first 128 bits of MSK or rMSK are used as key derivation key, *K*.

— *L* - The binary length of derived keying material MISK. *L* is determined by the selected ciphersuite, which is specified in 9.2.3.

— *h* - The output binary length of PRF used in the key derivation. That is, *h* is the length of the block of the keying material derived by one PRF execution. Specifically, for HMAC-SHA-1, *h* = 160 bits; for HMAC-256, *h* =256 bits; for CMAC-AES, *h* = 128 bits.

— *n* - The number of iterations of PRF in order to generate *L*-bits keying material.

— *Nonce-T* and *Nonce-N* - The nonces exchanged during the execution of service access authentication.

— *c* - The ciphersuite code is a one octet string specified for each ciphersuite. The code is defined in

9.2.3.

— *v* - The length of the binary representation of the counter and the length of keying material *L*. The default value for *v* is 32.

— “MISK” - 0x4D495354, ASCII code in hex for string “MISK.”

— *[a]2* - Binary representation of integer *a* with a given length.

For a given PRF, the key derivation for MISK can be described in the following procedures:

**Fixed input values**: *h* and *v*.

**Input**: *K*, *Nonce-T*, *Nonce-N*, *L*, and ciphersuite code.

**Process**:

a) *n* :=

*L* ⁄ *h* ;

b) If *n > 2v -1*, then indicate an error and stop. c) *Result (0)* := empty string.

d) For *i = 1* to *n*, do

i) *K(i) := PRF(K, “MISK” || [i]2 || Nonce-T || Nonce-N || c || [L]2)*. ii) *Result(i) = Result (i-1) || K(i)*.

e) Return *Result (n)* and *MISK* is the leftmost *L* bits of *Result (n)*. Output: *MISK*.

The *MISK* is parsed in such a way that

*MISK = MIAK || MIIK || MIEK*. (1) With the above procedure, a key hierarchy is derived as shown in Figure 36.

MSK or rMSK

K

MISK

|  |  |  |
| --- | --- | --- |
| MIAK | MIIK | MIEK |
|  | | |
|  | | |

**Figure 36—MIS Key Hierarchy**

**9.2.3 EAP-generated MIS security association**

When an MIS SA is established through an EAP method with key establishment, the SA consists of the keys, the key derivation functions, and the ciphersuite. The key derivation functions, encryption algorithms, and integrity algorithms are specified in Table 24.

**Table 24—Cryptographic algorithms**

|  |  |
| --- | --- |
| **Encryption algorithm** | **Description** |
| AES\_CBC | AES CBC mode ([NIST SP 800-38A]) |
| NULL | No encryption is applied |
| **Integrity algorithm** | **Description** |
| HMAC\_SHA1\_96 | HMAC-SHA1 with 96 bits MAC ([FIPS  198]) |
| AES\_CMAC | AES CMAC mode with 128 bits MAC ([NIST SP 800-38B]) |
| **Authenticated encryption** | **Description** |
| AES\_CCM | AES-CCM mode ([NIST SP 800-38C]) |
| **PRF used for key derivation function** | **Description** |
| PRF\_CMAC\_AES | AES CMAC as PRF in counter mode  ([NIST SP 800-108]) |
| PRF\_HMAC\_SHA1 | HMAC-SHA1 as PRF in counter mode  ([NIST SP 800-108]) |
| PRF\_HMAC\_SHA256 | HMAC-SHA256 as PRF in counter mode  ([NIST SP 800-108]) |

The ciphersuites and key lengths are defined in Table 25.

**Table 25—Ciphersuites**

|  |  |  |  |
| --- | --- | --- | --- |
| **Code** | **Encryption algorithm** | **Integrity algorithm** | **MISK length (L)** |
| 00000010 | AES\_CBC | HMAC\_SHA1\_96 | 384 |
| 00000100 | NULL | HMAC\_SHA1\_96 | 256 |
| 00000101 | NULL | AES\_CMAC | 256 |
| 00000110 | AES\_CCM | | 256 |

A default ciphersuite is defined as AES\_CCM. The default PRF is defined as PRF\_CMAC\_AES. The protection mechanisms for MIS messages are defined in 9.3.

**9.2.4 Termination**

A termination phase is defined as a mechanism to allow either an MN or a PoS to release the resource such as keys, authorized service access, etc. obtained through a service access authentication. Termination shall take place by either of two mechanisms:

a) Termination messages: These messages allow one party to explicitly inform another party the current authentication status is terminated. This option is supported by MIS\_Termination\_Auth messages defined in 8.6.1.14 and 8.6.1.15.

b) State timeout: A lifetime is defined for an MIS SA. After the time period defined by the lifetime, the MIS SA is terminated. The lifetime of the SA must be no longer than the MSK or rMSK lifetime, and communicated to the MIS node acting as the EAP peer by the Lifetime TLV included in MIS\_Auth request and MIS\_Auth response messages defined in 8.6.1.12 and 8.6.1.13.

**9.3 MIS message protection mechanisms for EAP-generated SAs**

**9.3.1 MIS\_Auth message protection**

The MIS\_Auth messages are not protected using the MIS SA established after a successful media independent service access authentication. MIS\_Auth messages are integrity protected by including an AUTH TLV generated using MIAK derived from the MSK or rMSK, as described in 9.2.2, with a PRF. The AUTH TLV value is generated as follows:

*AUTH TLV value = PRF(K, "AUTH-TLV" | MIS\_Auth message| MNCiphersuite | PoSCiphersuite),*

where

— K -MIAK

— “AUTH-TLV”- 0x415554482D544C56, ASCII code in hex for string “AUTH-TLV”

— MIS\_Auth message - an MIS\_Auth message in which AUTH TLV filled with 0s

— MNCiphersuite - Ciphersuite TLV sent by the MN

— PoSCiphersuite - Ciphersuite TLV sent by the PoS

PRF function is one of the following as negotiated:

a) PRF\_CMAC\_AES

b) PRF\_HMAC\_SHA1

c) PRF\_HMAC\_SHA256

The PRF output length must be truncated to 128 bits. If the PRF output length is more than 128 bits, the 128 leftmost bits of the output must be used as the AUTH TLV value.

The PRF output length must be truncated to 128 bits. If the PRF output length is more than 128 bits, the 128 leftmost bits of the output must be used as the AUTH TLV value.

**9.3.2 MIS PDU protection procedure**

Depending on the selected ciphersuite, an MIS PDU may be encrypted, integrity protected, or protected in both aspects. Correspondingly, an SA may identify an encryption key (MIEK), an integrity key (MIIK), or both MIEK and MIIK. When both encryption and integrity protection are applied, they may be accom- plished by two algorithms such as AES in CBC mode and HMAC-SHA1\_96 or by one authenticated encryption algorithm such as AES in CCM mode.

The portion to be protected in an MIS message includes the MIS service specific TLVs. That is, the source MISF identifier TLV and the destination MISF identifier TLV are not protected. The protection is applied based on an SA. An SAID TLV is carried in place of source and destination MISF identifier TLVs except for the case of transport address change where both an SAID TLV and source and destination MISF identi- fier TLVs are carried as described in 8.4.1a. An example procedure is illustrated in Figure 37.

When fragmentation is applied to an MIS PDU, then instead of service specific TLVs, the data to be pro- tected comprise a fragment payload. The values in the header fields M (more fragment) and FN (fragment number) are the same after the protection.

SA MIS header

(S=0)

Source MISF Identifier TLV

Destination MISF Identifier TLV

Service Specific TLVs or a fragment

yes

no

Encryption applied?

Service Specific

TLVs

yes

AES-CCM?

no

Service Specific

TLVs

yes

Integrity protected? no

AES-CCM

Encryption

Encrypted Service Specific TLVs

Service Specific

TLVs

Encrypted Service Specific TLVs

Security TLV

MIC

MIC Algorithm

Encrypted

MIC Algorithm

Fail

Service Specific

TLVs

MIC

Security TLV

Plaintext Service Specific TLVs

MIC

Security TLV

Set S bit in the Header and add SAID TLV

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
| MIS header  (S=1) | Source MISF Identifier TLV | Destination MISF Identifier TLV | SAID TLV | Security TLV |
|  | | | | |

**Figure 37—MIS PDU protection procedure**

**9.3.3 MIS PDU protection by AES-CCM**

AES in CCM mode as specified in NIST SP 800-38C shall be the default ciphersuite. The parameters used in AES-CCM, the nonce construction, the operational procedures, and the security TLV under AES-CCM protection shall be set according to the rules given in 9.3.3.1 through 9.3.3.3.

**9.3.3.1 AES-CCM Parameters**

For AES-CCM the following parameter values shall be set:

a) *t* - The length of MIC is 12 octets (96 bits).

b) *n* - The length of the nonce *N* is 13 octets (104 bits).

c) *q* - The length of the binary representation of the octet length of the data to be encrypted is 2 octets

(16 bits).

**9.3.3.2 Construct AES-CCM Nonce**

AES-CCM uses a nonce to construct an initialization vector and also the counter. CCM requires a unique nonce value for each MIS message protected by a given MIEK. In this standard, the nonce is 13 octets and consists of the following three portions.

a) Transaction ID (12 bits, from the MIS header) plus 4 reserved bits (set to zero);

b) Sequence number (10 octets, denoted as *SN0*, *SN1*, ..., *SN9*) starting from all zeros; and c) FN (7 bits, from the MIS header) plus 1 reserved bit (set to zero).

The nonce construction is illustrated in Figure 38.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Transaction ID (12) | Resv  (4) | *SN0*, *SN1* ..., *SN9*  (80) | FN (7) | Resv  (1) |

**Figure 38—AES-CCM Nonce Construction**

The *SN* is increased by a positive number for each MIS PDU. The SN shall never repeat for a series of encrypted MIS PDUs using the same MIEK. For a given SA, each of MISFs keeps an SN, which is the highest as used for a given MIEK.

**9.3.3.3 Operational procedures in AES-CCM**

**9.3.3.3.1 Encapsulation**

For a given SA, the prerequisites for AES-CCM encapsulation includes an encryption key MIEK, an AES block cipher encryption block, and the values of parameters *t*, *n*, and *q*. The plaintext, *P,* to be encrypted and authenticated is formed by concatenating all the service specific TLVs as presented in MIS PDU with the padding. In this standard, the associated data, A, is null. The data, *P,* is partitioned with necessary padding to

16-octet blocks *B1*, *B2*, ..., *Br* as specified in SP 800-38C. The octet block, *B0,* is an initialization vector and formed with 1-octet flags, 13-octet nonce *N*, and 2-octet integer *Q*, where *Q* is the octet length of *P*. The format of *B0* is illustrated in Figure 39

Flags

(1 octet)

Nonce

(13 octets)

*Q*

(2 octets)

**Figure 39—Format of *B0***

The flags are formed by the following data:

— 1 reserved bit, which is set to zero;

— 1 bit flag for the associated data, which is zero;

— 3 bits to represent (*t*-2)/2, which is 101 (*t* =12);

— 3 bits to represent *q*-1, which is 001 (*q* = 2).

The counter *Ctr(i)*, *i* = 0, 1, ..., r, is formed with 1-octet flags; 13-octet nonce *N*; and 2-octet integer *i.* The format of *Ctr(i)* is illustrated in Figure 40

Flags

(1 octet)

Nonce

(13 octets)

*i*

(2 octets)

**Figure 40—Format of Counter *Ctr(i)***

The flags for *Ctr(i)* is 00000001.

The encapsulation of an MIS PDU consists of the following steps:

a) Fetch Transaction ID and FN from the MIS header. b) Increment a positive number of *SN* to update the SN. c) Construct the nonce, *N*, as described in 9.3.3.2.

d) Input *N* and *P* to AES-CCM generation-encryption process as specified in SP 800-38C. The *B0* and all the counter numbers are formed as described in Figure 39 and Figure 40, respectively.

e) Obtain the output, C, of AES-CCM.

**9.3.3.3.2 Decapsulation**

For a given SA, the prerequisites for AES-CCM decapsulation includes an encryption key MIEK, AES

block cipher encryption block, the parameters *t*, *n*, and *q*.

The decapsulation of a protected MIS PDU consists of the following steps. a) Fetch Transaction ID and FN from the MIS header.

b) Fetch *SN* from the security TLV.

c) Construct the nonce, *N,* as described in 9.3.3.2.

d) Input *N* and *C* to AES-CCM decryption-verification process as specified in SP 800-38C. The *B0* and all the counter numbers are formed as described in 9.3.3.3.1.

e) Obtain the output, *P,* or “INVALID”.

**9.3.3.4 Format of security TLV**

The ENCR\_BLOCK data of the Security TLV in a protected MIS message with AES-CCM is formed by SN and ciphertext C, which is the ciphertext of P and T (MIC). It is illustrated in Figure 41. Since MIC is carried in the ENCR\_BLOCK data, the INTEGR\_BLOCK in MIS\_SPS\_RECORD is not chosen for AES- CCM.

SN

(10 octets)

C (Encryption of P and T) (Q + 12 octets)

ENCR\_BLOCK

**Figure 41—Security TLV for AES-CCM**

**9.3.4 MIS PDU protection by AES in CBC mode and HMAC-SHA1-96**

This ciphersuite includes two algorithms, encryption algorithm AES CBC and message authentication code algorithm HMAC-SHA1-96. When an MIS PDU is protected, encryption is applied first and an MIC is generated on the ciphertext. The MIC is 12 octets (96 bits). In order to use the ciphersuite, two keys (MIIK and MIEK) are used for encryption/decryption and generation/verification of MIC respectively.

**9.3.4.1 Initialization vector for AES in CBC mode**

Encryption using AES in CBC mode needs an initialization vector, IV, of 16 octets (128 bits), *IV* = (*IV0*, *IV1*, ..., *IV15*). It can be selected randomly when encryption is executed. It is also needed for decryption. Therefore, for each protected MIS PDU, an IV is included in ENCR\_BLOCK as a part of security TLV.

**9.3.4.2 Operational procedures in applying AES CBC and HMAC-SHA1-96**

**9.3.4.2.1 Encapsulation**

The encapsulation of an MIS PDU includes the following steps:

a) Select a 16-octet initialization vector, *IV*.

b) Pad the plaintext, *P,* to a length of a multiple of 16 octets (128 bits) so that the padded plaintext can be represented as in *n* blocks *P0*, *P1*, .., *Pn-1*, each of which is 16 octets.

c) Apply AES CBC mode with *IV* and key, MIEK, on *P0*, *P1*, ..., *Pn-1* to obtain ciphertext *C0*, *C1*, ...,

*Cn-1*.

d) Input *M* = *IV*||*C0*||*C1*||...||*Cn-1*, where “||” means concatenating, as the message and MIIK as the key to HMAC-SHA1. Here padding may be needed to make the input message length to be a multiple 64 octets (512 bits). The most significant 12 octets of the output of HMAC-SHA1 is the MIC.

e) Output *C0*, *C1*, ..., *Cn-1* and MIC.

**9.3.4.2.2 Decapsulation**

The decapsulation of a protected MIS PDU includes the following steps:

a) Fetch the ciphertext *C0*, *C1*, ..., *Cn-1* and the initialization vector, *IV*, from ENCR\_BLOCK of secu- rity TLV.

b) Fetch the MIC from the INTG\_BLOCK of security TLV.

c) Input *M* = *IV*||*C0*||*C1*||...||*Cn-1*, where “||” means concatenating, as the message and MIIK as the key to HMAC-SHA1. Here padding may be needed to make the input message length a multiple 64 octets (512 bits). Compare the most significant 12 octets of the output of HMAC-SHA1 with the MIC. If they are identical, go to the next step. Otherwise, output “INVALID”.

d) Input ciphertext, *C0*, *C1*, ..., *Cn-1*, and MIEK to AES CBC mode to obtain plaintext, *P0*, *P1*, ..., *Pn-*

*1*.

e) Remove the padding if it is applied to obtain the plaintext, *P*. f) Output *P*.

**9.3.4.3 Format of security TLV**

When an MIS PDU is protected by AES in CBC mode and HMAC-SHA1-96, both ENCR\_BLOCK and INTG\_BLOCK appear in the security TLV. The initialization vector IV and the ciphertext, *C0*, *C1*, ..., *Cn-1*, are included in ENCR\_BLOCK and MIC is in INTG\_BLOCK as shown in Figure 42.

IV

(16 octets)

C0, C1, ..., Cn-1 (16n octets)

ENCR\_BLOCK

MIC

(12 octets)

INTG\_BLOCK

**Figure 42—Security TLV for AES CBC and HMAC-SHA1-96**

**9.3.5 MIS PDU protection by HMAC-SHA1-96**

This ciphersuite includes one message authentication code algorithm, HMAC-SHA1-96. It generates a 12 octets (96 bits) MIC over the protected data using key MIIK.

**9.3.5.1 MIC generation and verification**

**9.3.5.1.1 MIC generation**

A MIC is generated in the following steps:

a) The data, *P*, to be protected is padded to a length of a multiple of 64 octets (512 bits). b) Input the padded data and the key, MIIK, to HMAC-SHA1.

c) Obtain output of HMAC-SHA1.

d) Truncate the output of HMAC-SHA1 to obtain the most significant 96 bits as the MIC. e) Output MIC.

**9.3.5.1.2 MIC verification**

A MIC is verified in the following steps:

a) Fetch the data, *P*, from the ENCR\_BLOCK of security TLV. Pad it to a length of a multiple of 64 octets (512 bits).

b) Fetch the MIC from INTG\_BLOCK of security TLV.

c) Input the padded data and the key, MIIK, to HMAC-SHA1. d) Obtain output of HMAC\_SHA1.

e) Compare the most significant 96 bits of the output of HMAC-SHA1 with MIC. f) If they are identical, output “VALID”; Otherwise, output”INVALID”.

**9.3.5.2 Format of security TLV**

When an MIS PDU is protected by HMAC-SHA1-96, the plaintext is included in ENCR\_BLOCK, even though it is not encrypted and the MIC is in the INTG\_BLOCK as shown in Figure 43.

Plaintext

MIC

(12 octets)

ENCR\_BLOCK

INTG\_BLOCK

**Figure 43—Security TLV for HMAC-SHA1-96**

**9.3.6 MIS PDU protection by AES-CMAC**

This ciphersuite includes one MAC algorithm, AES-CMAC. It generates a 12 octets (96 bits) MIC over the protected data using key, MIIK.

**9.3.6.1 MIC generation and verification**

**9.3.6.1.1 MIC generation**

A MIC is generated in the following steps:

a) Input the data, *P*, to be protected and the key, MIIK, to AES-CMAC. (For AES-CMAC, the padding is specified as a part of the algorithm.)

b) Obtain output of AES-CMAC.

c) Truncate the 16 octets (128 bits) output of AES CMAC to obtain the most significant 96 bits as the

MIC.

d) Output MIC.

**9.3.6.1.2 MIC verification**

A MIC is verified in the following steps:

a) Fetch the data, *P*, from the ENCR\_BLOCK of security TLV. b) Fetch the MIC from INTG\_BLOCK of security TLV.

c) Input the data, *P*, to be protected and the key, MIIK, to AES-CMAC. (For AES-CMAC, the padding is specified as a part of the algorithm.)

d) Obtain output of AES-CMAC.

e) Compare the most significant 96 bits with the MIC.

f) If they are identical, output “VALID”; Otherwise, output “INVALID”.

**9.3.6.2 Format of security TLV**

When an MIS PDU is protected by AES-CMAC, the plaintext is included in the ENCR\_BLOCK, even though it is not encrypted and the MIC is in the INTG\_BLOCK as shown in Figure 44.

Plaintext

MIC

(12 octets)

ENCR\_BLOCK

INTG\_BLOCK

**Figure 44—Security TLV for AES-CMAC**

**9.4 Common procedures**

The following procedures are common for both (D)TLS- and EAP-generated MIS SAs.

**9.4.1 Sending**

For sending a remote MIS message in a protected manner, an MIS PDU is created in the following steps:

a) At the sender, which can be an MN or a PoS, the MIS user generates an MIS primitive and passes it to the MISF.

b) The MISF at the sender constructs an MIS PDU. If an MIS security association (SA) exists, then the MISF at the sender applies (D)TLS protection algorithms specified by the negotiated ciphersuite in the handshake to the MIS PDU and then encapsulates the protected MIS PDU in a security TLV. If no MIS SA exists, then the MIS PDU is passed to the transport protocol of the MIS message.

c) The security TLV is encapsulated in an MIS PDU with the S bit in the MIS header set to one. d) The MISS PDU is passed to the transport protocol of the MIS message.

**9.4.2 Receiving**

For receiving a protected MIS message from a remote entity, the protected MIS PDU is processed in the following steps:

a) At the receiver, which can be an MN or a PoS, the MISF receives a protected MIS PDU from the transport protocol of the MIS message.

b) If the S bit is set to one in the header, the MISF processes security TLV and extract the plaintext

MIS PDU. Otherwise, it takes MIS PDU as is.

c) The MISF creates an MIS primitive from MIS PDU and passes it to the MIS user at the receiver.

The processing steps at the sender and receiver are described in Figure 45.

MIS User MIS User

MISF

MIS Primitive

MIS PDU

MISF

MIS Primitive

MIS PDU

Apply protection mecha- nisms and generate security TLV as payload to MIS PDU

Apply de-protection mecha- nisms and obtain MIS PDU from security TLV

L2 or L3 Transport L2 or L3 Transport

**Figure 45—Sending and Receiving Protected MIS PDU**

The transport protocol entities to be associated with an MIS SA are MISF peers and are identified by MISF identifiers. Therefore, the transport address of an MISF can change over the lifetime of an MIS SA as long as the mapping between the transport address and the MISF identifier of an MISF is maintained.

**10. Proactive authentication**

In a handover from a service PoA to a target PoA, a MN may need to authenticate to the target network through an authentication mechanism required by the target network. This clause specifies the mechanisms to use MIS to assist proactive authentications to reduce the latency due to media access authentication and key establishment.

This standard introduces two options to conduct the proactive authentication with a targeted network. The first option is called unbundled media access proactive authentication. In such a proactive authentication, an MN conducts an authentication with the targeted network as it is required for accessing that network through a specific media. In this case, the authenticator is a media specific authenticator (MSA). The authentication messages are passed between the MN and the MSA through an MIS PoS. The authentication messages between the MN and the PoS are carried through MIS messages. The second option is to bundle the media access proactive authentication to the MIS service access authentication. In this case, at the end of MIS service access authentication, the MN and the PoS also establish a key(s) for a target PoA(s). The key(s) are distributed to the PoA(s) so that when a handover to one of the PoAs happens, the MN can establish a protected link with the PoA. The MIS message exchange between an MN and a PoS is common to both bundled and unbundled proactive authentication. The only difference is that the bundled proactive authentication uses a key established through MIS service access authentication. The MIS message exchange for bundled and unbundled proactive authentication is described in 10.1.

**10.1 Media specific proactive authentication**

In a media access proactive authentication, a PoS passes authentication messages between the MN and a media specific authenticator (MSA). The protocol stacks in each interface are illustrated in Figure 46. In scenarios where MSA/Target PoA is reachable via same media as MN and PoS, EAP messages received at PoS are directly forwarded to the target PoA.

EAP Peer / MN PoS MSA / Target PoA

Proactive Authentication (EAP messsage in media specific frame)

Proactive Authentication (EAP messsage in media specific frame)

Proactive Authentication (EAP messsage in media specific frame)

MIS Protocol

MIS Protocol

Network / Link Layer

Network/Link Layer

**Figure 46—Protocol Stack for MIS Supported Proactive Authentication**

**10.1.1 Procedures in a media specific proactive authentication**

An MIS assisted media specific proactive authentication includes the following main procedures.

**10.1.1.1 PoS and candidate media specific authenticator discovery**

Before an MN initiates an MIS assisted proactive authentication, the MN needs to know the PoS’s address and the candidate media specific authenticators’ link layer addresses. The corresponding candidate MSAs’ addresses can be discovered by using the information elements (IEs) specified in 6.5.4.

**10.1.1.2 Proactive authentication through EAP or ERP**

In order to execute a proactive authentication, the EAP or ERP messages are encapsulated in the extended MIS messages as L2 frames. When the PoS receives an encapsulated EAP or ERP message, it decapsulates it, then forwards it to the candidate media specific authenticator (MSA). The EAP or ERP messages are encoded as OCTET\_STRING.

**10.1.1.3 Media specific association handshake**

When the MN decides to handover to a candidate network, the MN and the PoA, which is associated with the MSA, perform a media specific association based on the keying material derived by the proactive authentication. For example, the media specific handshake could be a 4-way handshake as in an IEEE 802.11 network. A media specific handshake may further derive Media specific session keys to protect the communication between the MN and the PoA once it is attached to it.

**10.1.2 Proactive authentication message format**

When a proactive authentication is executed through EAP [RFC3748] or ERP [RFC5296], the EAP packets are carried by MIS messages. MIS primitives for the link layer frames are defined in 7.4.28 for proactive authentication. The messages are defined in 8.6.1.18 and 8.6.1.19. The MIS messages for proactive authen- tication shall be protected by an MIS SA.

**10.2 Bundling media access authentication with MIS service access authentication**

When the trust relationship between media specific network access provider and the MIS service provider allows, a proactive authentication can be optimized by bundling the media access authentication with an MIS service access authentication. In this case, at the end of a successful service access authentication, a PoS will derive not only keys for MIS message protection as defined in 9.2.2 but also a key called media specific root key (MSRK). This key will be further used to derive a key or keys called media specific pair- wise master keys (MSPMKs) to be used by a target PoA or PoAs.

**10.2.1 Media specific key derivation**

**10.2.1.1 Derivation of media specific root key (MSRK)**

After a successful service access authentication through EAP or ERP, a master session key (MSK) or a re- authentication MSK (rMSK) is generated in the MN and the PoS. The media specific root key (MSRK) is derived from MSK or rMSK.

For the media specific root key derivation, the following notations and parameters are used:

— *K* - key derivation key. It is truncated from a master session key (MSK) or re-authentication MSK (rMSK). The length of *K* is determined by the pseudorandom function (PRF) used for key derivation. If HMAC-SHA-1 or HMAC-SHA-256 is used as a PRF, then the full MSK or rMSK is used as key

derivation key, *K*. If CMAC-AES is used as a PRF, then the first 128 bit of MSK or rMSK is used as key derivation key, *K*.

— *h* - The output binary length of PRF used in the key derivation. That is *h* is the length of the block of the keying material derived by one PRF execution. Specifically, for HMAC-SHA-1, *h* = 160 bits; for HMAC-256, *h* =256 bits; for CMAC-AES, *h* = 128 bits.

— *Nonce-T* and *Nonce-N* - The nonces exchanged during the execution of service access authentication.

— “MSRK” - 0x4D53524B, ASCII code in hex for string “MSRK.” The MSRK derivation is described as follows:

**Input**: *K*, *Nonce-T*, *Nonce-N*.

**Process**:

a) *MSRK:= PRF(K, “MSRK” || Nonce-T || Nonce-N)*. b) Return *MSRK*.

**Output**: *MSRK*

The binary length of *MSRK* is *h*. Depending on the PRF used for the MSRK derivation, it can be 128 bits,

160 bits, or 256 bits. The MSRK is used to derive media specific pairwise master keys (MSPMK).

**10.2.1.2 Derivation of media specific pairwise master keys (MSPMKs)**

Each MSPMK is derived specifically for a PoA. For the media specific pairwise master key (MSPMK)

derivation, the following notations and parameters are used:

— *K*- key derivation key. It can be a full length of *MSRK* or a portion of *MSRK*. Specifically, the length of *MSRK* is *h* which is determined by the PRF used for key derivation. If in MSRK derivation and in MSPMK derivation, the same PRF is used, the MSPMK derivation will be able to use the full length *MSRK*. However, in case that HMAC-SHA1 or HMAC-SHA256 is used in MSRK derivation, but CMAC-AES is used in MSPMK derivation, then only the first 128 bits of MSRK is used as a key derivation key in the MSPMK derivation.

— *MN\_LINK\_ID* - A link layer identity of the MN.

— *PoA\_LINK\_ID* - A link layer identity of a target point of attachment (PoA).

— “*MSPMK*” - 0x4D53504D4B, ASCII code in hex for string “MSPMK”. The MSPMK derivation is described as follows:

**Input**: *K*, *MN\_LINK\_ID*, *PoA\_LINK\_ID*.

**Process**:

a) *MSPMK:= PRF(K, “MS-PMK” || MN\_LINK\_ID|| PoA\_LINK\_ID)*. b) Return *MSPMK*.

**Output**: *MSPMK.*

The binary length of *MSPMK* is *h*. Depending on the PRF used for the above MSPMK derivation, it can be

128 bits, 160 bits, or 256 bits.

The new key hierarchy is illustrated in Figure 47.

MSK or rMSK

K

MSRK

|  |  |  |
| --- | --- | --- |
| MIAK | MIIK | MIEK |
|  | | |
| MISK | | |

MSPMKa MSPMKb

**Figure 47—Key Hierarchy for Bundle Case**

**10.2.2 Media specific key distribution**

Each MSPMK will be distributed to a PoA. The key distribution from the PoS to a PoA can be done through push or pull key distribution. In general, key distribution from a PoS to a PoA is out of the scope of this standard. However, MIS service can be used to trigger the key distribution. The key distribution can be triggered in the following methods.

**10.2.2.1 Push key distribution**

The objective of push key distribution is to trigger a PoS to push a key into a target PoA. To perform the installation, the MN uses the MIS protocol, which at this point could be protected, to notify the PoS to start the key installation. In the PoS, the key is pushed from MISF to a MIS user for the further distribution to a PoA. The primitives for push key distribution are defined in 7.4.27. The messages are defined in 8.6.1.16 and 8.6.1.17.

**10.2.2.2 Reactive pull key distribution**

A reactive pull key distribution is performed after the MN moves to the target PoA. Since no MIS function is used, this is out of the scope of this standard.

**10.2.2.3 Optimized proactive pull key distribution**

This mechanism allows the MN to perform a media-specific authentication proactively with a target PoA without being directly connected to the wireless link of the target PoA by means of sending link-layer frames through the PoS to the target PoA. The key hierarchy shared between the MN and the PoS is used in order to derive a pre-shared key to conduct a proactive authentication. The PoS is acting as a local authentication server (AS). The PoA receiving the link-layer frames with the authentication information can contact with the AS (the PoS) using the identifier provided during the service access authentication. Once the proactive authentication is completed, a media specific master session key (MSK) is distributed from the PaS (acting as an AS) to the PoA At the end, the MN and the PoA share the same media specific MSK. To perform this key distribution mechanism the primitives are defined in 7.4.28 and MIS messages are defined in 8.6.1.18 and 8.6.1.19.

.

# (informative) Bibliography

1. 3GPP TR 43.901, Feasibility Study on Generic Access to A/Gb Interface.
2. 3GPP TS 23.002, Network Architecture.
3. 3GPP TS 23.060, General Packet Radio Service (GPRS); Service Description; Stage 2.
4. 3GPP TS 23.234, 3GPP system to Wireless Local Area Network (WLAN) inter-working; System description.
5. 3GPP TS 24.007, Mobile Radio Interface Signaling, Layer 3, General Aspects.
6. 3GPP TS 25.331, Radio Resource Control Specification.
7. 3GPP2 C.S0001-D (2004-02), Introduction to cdma2000 Standards for Spread Spectrum Systems.
8. 3GPP2 C.S0002-D (2004-02), Physical Layer Standard for cdma2000 Spread Spectrum Systems.
9. 3GPP2 C.S0003-D (2004-02), Medium Access Control (MAC) Standard for cdma2000 Spread Spec­trum Systems.
10. 3GPP2 C.S0005-D (2004-02), Upper Layer (Layer 3) Signaling Standard for cdma2000 Spread Spec­trum Systems.
11. 3GPP2 C.S0006-D (2004-02), Analog Signaling Standard for cdma2000 Spread Spectrum Systems.
12. 3GPP2 C.S0024-A (V. 2.0, 2005-07), cdma2000 High Rate Packet Data Air Interface Specification.
13. 3GPP2 C.S0063-A (V. 1.0, 2006-04), cdma2000 High Rate Packet Data Supplemental Services.
14. 3GPP2 S.R0087, cdma2000 – WLAN Interworking.
15. 3GPP2 X.S0028-1 00 (V. 1.0, 2006-08), Wireless Local Area Network (WLAN) Interworking, Access to internet.
16. 3GPP2 X.S0028-200 (V. 1.0), Wireless Local Area Network (WLAN) Interworking, Access to Oper­ator Services and Mobility.
17. IEEE 100, *The Authoritative Dictionary of IEEE Standards Terms*, Seventh Edition.
18. IEEE Std 802.1AB™-2005, IEEE Standard for Local and Metropolitan Area Networks—Station and Media Access Control Connectivity Discovery.
19. IEEE 802.1ag™-2007, IEEE Standard for Local and Metropolitan Area Networks—Virtual Bridged Local Area Networks—Amendment 5: Connectivity Fault Management.
20. IEEE P802.21™ (Document 21-04-0087-12-0000, 2004-09), Media Independent Handover Ser­vice—Draft Technical Requirements.
21. IEEE P802.16g™-2007, IEEE Standard for Local and metropolitan area networks—Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems—Amendment 3: Management Plane Procedures and Services.
22. IETF RFC 791 (198 1-09), DARPA Internet Program Protocol.
23. IETF RFC 3629 (2003-11), UTF-8, a transformation format of ISO 10646.
24. IETF RFC 3753 (2004-06), Mobility Related Terminology.
25. IETF RFC 4291 (2006-02), IP Version 6 Addressing Architecture.
26. IETF RFC 4443 (2006-03), Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification.
27. IETF RFC 5184 (2008-05), Unified L2 Abstractions for L3-Driven Fast Handover.
28. IETF Internet Draft (draft-ietf-geopriv-radius-lo-19.txt, 2008-01), Carrying Location Objects in RADIUS and Diameter.
29. IETF Internet Draft (draft-ietf-mipshop-mos-dhcp-options-03.txt, 2008-06), Dynamic Host Configu­ration Protocol (DHCPv4 and DHCPv6) Options for Mobility Server (MoS) discovery.
30. IETF Internet Draft (draft-ietf-mipshop-mos-dns-discovery-01 .txt, 2008-05), Locating Mobility Serv­ers using DNS.
31. IETF Internet Draft (draft-ietf-mipshop-mstp-solution-04.txt, 2008-05), Mobility Services Frame­work Design.
32. ITU-T Recommendation X.210 (11/93), Information technology-Open systems interconnection-Basic Reference Model: Conventions for the definition of OSI services (common text with ISO/IEC 10731).
33. ITU-T Recommendation X.690, Information technology—ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).
34. ITU-T Recommendation Y. 1541, Network performance objectives for IP-based services.
35. ITU-T Recommendation Z. 120 (2004), Programming Languages-Formal Description Techniques (FDT)-Message Sequence Chart (MSC).
36. W3C Recommendation, OWL Web Ontology Language Reference.
37. W3C Recommendation, RDF Vocabulary Description Language 1.0: RDF Schema.
38. W3C Recommendation, SPARQL Query Results XML Format.

# (normative) Quality of service mapping

This annex provides the mapping between QoS parameters with various technologies. A flow diagram is provided that shows the setting and reporting of QoS parameters using the standard IEEE 802.21 primitives. Table B. 1, Table B.2, and Table B.3 show the mapping between generic QoS parameters and those used by different technologies such as IEEE 802.11, IEEE 802.16, and 3GPP. B.3 describes how the generic QoS parameters can be derived from the access link specific parameters.

A transmitted packet over a communication medium can experience the following outcomes:

* Be received with no errors at its intended destination
* Be received with errors at its intended destination
* Not be received in which case it is said that the packet is lost

A communication medium represents one or multiple point-to-point network segments that are termed links in this standard.

The maximum attainable speed of information transfer over a given communication channel can be constant, as is usually the case with communication channels involving only wired links, or it can be time varying at different scales, as is often the case for communication channels involving wireless links. This measure will be called link throughput, for the purposes of this standard.

The ability of the link to provide accurate information transfer can be described via a statistical model characterized by the following parameters:

* Minimum Packet Transfer Delay: is defined as the minimum delay over a population of interest.
* Average Packet Transfer Delay: is defined as the arithmetic mean of the delay over a population of interest.
* Maximum Packet Transfer Delay: is defined as the maximum delay over a population of interest.
* Jitter: is defined as the standard deviation of the delay over a population of interest.
* Packet Loss Rate: is defined as the ratio between the number of frames that are transmitted but not received and the total number of frames transmitted over a population of interest.
* Packet Error Rate: is defined as the ratio between the number of packets that have been received with errors and the total number of packets present in a population of interest. Note that if the link supports re-transmission, then the Packet Error Rate includes it, otherwise it does not include it.

For a link that supports CoS differentiation, per CoS traffic accuracy parameters need to be maintained in order to provide insights on how individual traffic classes are faring.

In summary, the following set of parameters characterizes the speed and accuracy of the information transfer that a multi-CoS traffic link supports:

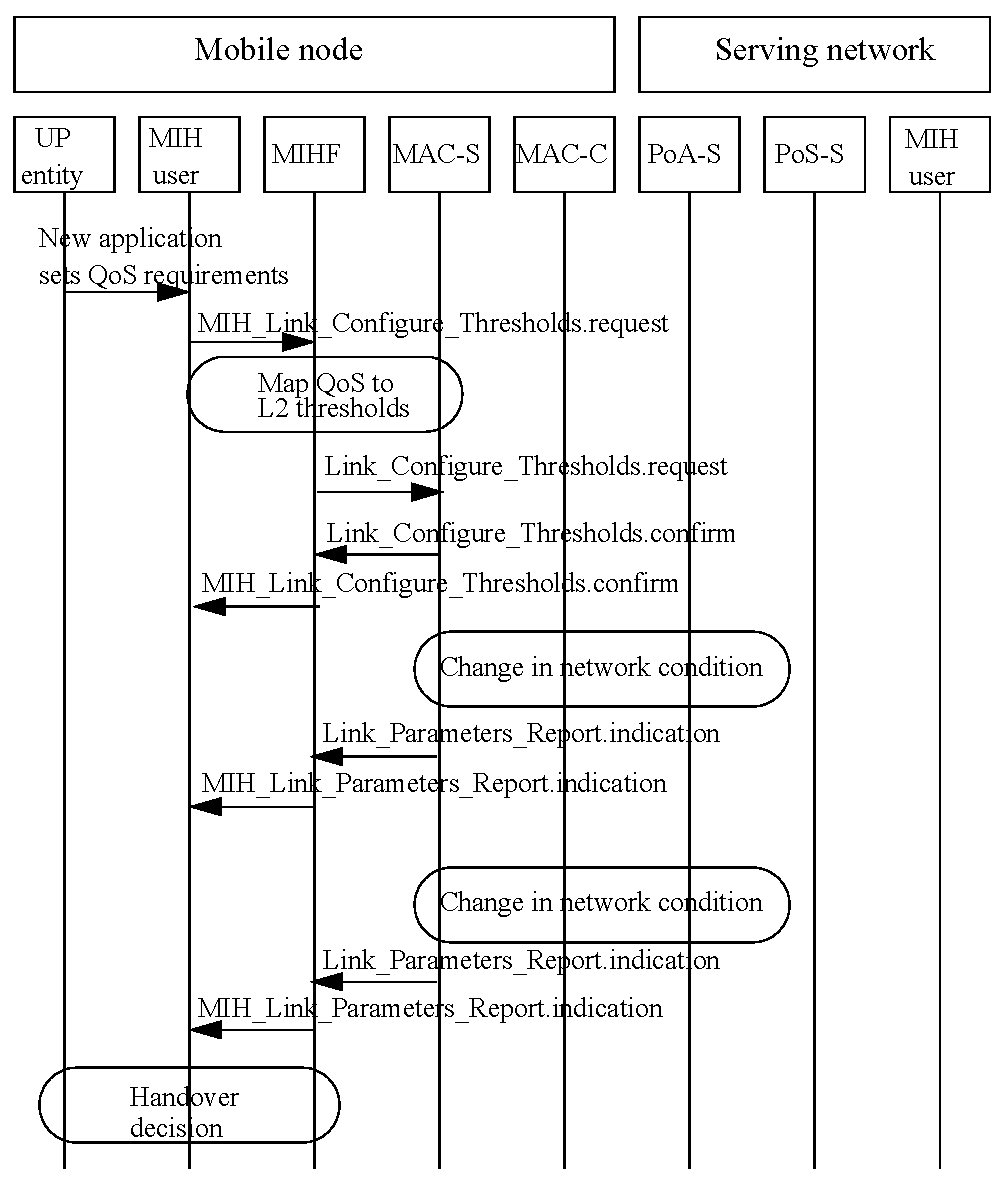
1. Link Throughput, the number of bits successfully received divided by the time it took to transmit them over the medium.
2. Link Packet Error Rate: representing the ratio between the number of frames received in error and the total number of frames transmitted in a link population of interest.
3. Supported Classes of Service: represents the maximum number of differentiable classes of service supported by this link.
4. Class of Service Parameters List: For each of the supported classes of service the following parameters are defined:
   1. Class Minimum Packet Transfer Delay: is defined as the minimum delay over a class population of interest.
   2. Class Average Packet Transfer Delay: is defined as the arithmetic mean of the delay over a class population of interest.
   3. Class Maximum Packet Transfer Delay: is defined as the maximum delay over a class population of interest.
   4. Class Packet Delay Jitter: is defined as the standard deviation of the delay over a class population of interest.
   5. Class Packet Loss Rate: is defined as the ratio between the number of frames that are transmitted but not received and the total number of frames transmitted over a class population of interest.

## Generic IEEE 802.21 QoS flow diagram

Figure B.1 represents an example flow diagram for using the QoS framework defined by the MISF. The following terms are used in Figure B.1:

* UP Entity: An upper layer entity such as a multimedia application;
* MAC-S: The MAC layer of the interface that is currently serving the MN;
* MAC-C: The MAC layer of an interface that is not currently serving the MN;
* PoA-S: The serving PoA;
* PoS-S: The serving PoS.

The MIS\_Link\_Configure\_Thresholds primitive is used to set the application quality of service requirements and make it available to the MISF. These parameters are mapped into media-specific measurements at the MIS layer and then used to configure the link parameter thresholds. While this mapping is not defined by other standards, Table B.1 and Table B.2 provide such mappings. The MIS\_Link\_Parameters\_Report primitive is used to relay link specific measurements back to the MIS user.



**Figure B.1—An example flow for setting application QoS requirements**

## Generic IEEE 802.21 QoS parameter mappings

The tables provide mappings of the standard IEEE 802.21 QoS parameters to the access link technology  
specific parameters. Table B. 1 is informative and list measurements defined in IEEE Std 802.11 that may be used in computing the QOS performance metrics defined in this document. For IEEE 802.11, a collection of the QoS parameters can be on an individual station measurement basis, since this is a media using a distributed (symmetric) access technology.

**Table B.1—QoS parameter mapping for IEEE 802.11**

|  |  |  |  |
| --- | --- | --- | --- |
| **IEEE 802.21 link QoS parameters** | **Related IEEE 802.11 parameters** | **IEEE 802.11 IE name** | **Note** |
| Throughput | Not currently supported. |  | Measurement is defined as the total number of octets transmitted / Mea­surement duration.” |
| Packet error rate | TransmittedFragmentCount MulticastTransmittedFrameCount FailedCount  ReceivedFragmentCount  (See NOTE) MulticastReceivedFrameCount FCSErrorCount (See NOTE) TransmittedFrameCount  RetryCount  MultipleRetryCount FrameDuplicateCount RTSSuccessCount RTSFailureCount ACKFailureCount | STA Statistics Report |  |
| Supported number of COS | 4 for IEEE 802.11e, 8 for HCCA, 1 for non-IEEE 802.1 1e systems |  |  |
| CoS minimum packet transfer delay | Transmit Delay Histogram (See NOTE) | Transmit Stream/ Category Mea- surement Report | Trigger (Option) (only to specific STA) |
| CoS average packet trans- fer delay | Average Transmit Delay | Transmit Stream/ Category Mea- surement Report | Trigger (Option) (only to specific STA) |
| CoS maximum packet transfer delay | Transmit Delay Histogram (See NOTE) | Transmit Stream/ Category Mea- surement Report | Trigger (Option) (only to specific STA) |
| CoS packet delay jitter | Transmit Delay Histogram  (See NOTE)  Average Transmit Delay (See NOTE) | Transmit Stream/ Category Mea- surement Report | Trigger (Option) (only to specific STA) |

**Table B.1—QoS parameter mapping for IEEE 802.11 *(continued)***

|  |  |  |  |
| --- | --- | --- | --- |
| **IEEE 802.21 link QoS parameters** | **Related IEEE 802.11 parameters** | **IEEE 802.11 IE name** | **Note** |
| CoS packet loss rate | QoSTransmittedFragmentCount (See NOTE)  QoSFailedCount  QoSRetryCount QoSMultipleRetryCount QoSFrameDuplicateCount QoSRTSSuccessCount QoSRTSFailureCount QoSACKFailureCount (See NOTE) QoSReceivedFragmentCount QoSTransmittedFrameCount QoSDiscardedFrameCount QoSMPDUsReceivedCount QoSRetriesReceivedCount | STA Statistics Report |  |
| Transmitted MSDU Count (See NOTE)  MSDU Discarded Count  MSDU Failed Count (See NOTE) MSDU Multiple Retry Count QoS CF-polls Lost Count | Transmit Stream/ Category Mea- surement Report | Trigger (Option) (only to specific STA) |
| NOTE—This parameter is most likely to be used to directly derive IEEE 802.21 LinkQoSParameters. See B.3 for example derivations. | | | |

Table B.2 and Table B.3 show example mappings for IEEE 802.21 QoS link parameters and other link specific parameters for IEEE 802.16, 3GPP, and 3GPP2. For these technologies control is usually by means of a base station, not an individual station, since the media is controlled using asymmetric access.

**Table B.2—QoS parameter mapping for IEEE 802.16 and 3GPP2**

|  |  |  |
| --- | --- | --- |
| **IEEE 802.21 link QoS parameters** | **IEEE 802.16** | **3GPP2** |
| Throughput | Maximum Sustained Traffic Rate | Peak\_Rate |
| Packet loss rate |  | Max\_IP\_Packet\_Loss\_Rate |
| Packet error rate | Packet Error Rate |  |
| CoS minimum packet transfer delay |  |  |
| CoS average packet transfer delay |  |  |
| CoS maximum packet transfer delay | Maximum Latency | Max\_Latency |
| CoS packet delay jitter | Tolerated Jitter | Delay\_Var\_Sensitive |

**Table B.3—QoS parameter mapping for 3GPP**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IEEE 802.21 link QoS parameters** | **Related 3GPP parameters** | | | |
| Supported number of CoS | 4 | | | |
| **Conversational** | **Streaming** | **Interactive** | **Background** |
| Throughput | Peak throughput | | | |
| Mean throughput | | | |
| Maximum bit rate for uplink/downlink | | | |
| Guaranteed bit rate for uplink/downlink | |  |  |
| Link packet error rate | SDU Error Ratio | | | |
| Residual Bit Error Rate | | | |
| CoS minimum packet transfer delay | Transfer delay | |  |  |
| CoS average packet  transfer delay | Transfer delay | |  |  |
| CoS maximum packet transfer delay | Maximum Transfer delay | |  |  |
| CoS packet transfer  delay jitter |  | Delay Variation |  |  |
| CoS packet loss rate | Residual Bit Error Rate | | | |
| SDU Error Ratio | | | |

## Deriving generic IEEE 802.21 QoS parameters

### General

The following subclauses describe how to derive generic QoS parameters from IEEE Std 802.11 link measurement parameters. This derivation relies on incremental values of counters as specified in the IEEE Std 802.11.

Note that the parameters are unicast parameters that are unrelated to multicast traffic.

### Packet loss rate

To calculate the packet loss rate (PLR), one uses Equation (1).

 (1)

According to IEEE Std 802.11, a packet is a MAC user data packet or MAC service data unit (MSDU).

The *PLRMSDU* can be derived from the **Transmit Stream/Category Measurement Report** using Equation (2).

 (2)

=

### Packet error rate

The packet error rate (PER) can be calculated using Equation (3).

 (3)

Unlike for PLR, this parameter is only defined for the IEEE 802.11 MPDU. The PER can be derived from the **STA Statistics Report information element** using Equation (4).

 (4)

### Average transfer delay

In IEEE Std 802.11k, the transmit delay (MSDU delay) is defined as follows:

Transmit delay (MSDU delay): The delay shall be measured from the time the MSDU is passed to the MAC sublayer until the point at which the entire MSDU has been successfully transmitted including receipt of the final ACK.

If the average MSDU transmit delay is used for the IEEE 802.21 average transfer delay, it can be derived from **Transmit Stream/Category Measurement Report.**



### Packet transfer delay jitter

Using the IEEE 802.21 definition of “the standard deviation of the delay over a population of interest,” the IEEE 802.11 MAC sublayer provides the **Transmit Stream/Category Measurement Report** and measurement parameters to calculate the standard deviation of delay.

* QoS Metric information element includes:

1. Transmit Delay Histogram
2. Average Transmit Delay parameters

Variance calculation using discrete density function is given as



Therefore, the packet transfer delay jitter for MSDU level is



where,

*N* = the number of bins of Transmit Delay Histogram

*Pi*= the value (measured percentile) of ith bin of Transmit Delay Histogram

*xi* = the mean value of the delay range of ith bin

# (to be excluded) (informative) Handover procedures

# (normative) Mapping MIS messages to reference points

Table D. 1 maps the MIS messages to the MIS communication model reference points.

Table D.1—Mapping MIS messages to reference points

|  |  |
| --- | --- |
| **MIS message name** | **Reference point** |
| MIS\_Capability\_Discover | RP1, RP2, RP3, RP4, RP5 |
| MIS\_Event\_Subscribe | RP1, RP3 |
| MIS\_Event\_Unsubscribe | RP1, RP3 |
| MIS\_Register | RP1, RP3, RP5 |
| MIS\_DeRegister | RP1, RP3, RP5 |
|  |  |
| MIS\_Link\_Detected | RP1, RP3 |
| MIS\_Link\_Up | RP1, RP3, RP5 |
| MIS\_Link\_Down | RP1, RP3, RP5 |
| MIS\_Link\_Parameters\_Report | RP1, RP3, RP5 |
| MIS\_Link\_Going\_Down | RP1, RP3, RP2 |
| MIS\_Link\_Handover\_Imminent | RP1, RP3, RP2 |
| MIS\_Link\_Handover\_Complete | RP1, RP3 |
|  |  |
| MIS\_Link\_Get\_Parameters | RP1, RP3, RP2 |
| MIS\_Link\_Configure\_Thresholds | RP1, RP3 |
| MIS\_Link\_Actions | RP1, RP3 |
| MIS\_Net\_HO\_Candidate\_Query | RP1, RP3 |
| MIS\_MN\_HO\_Candidate\_Query | RP1, RP3 |
| MIS\_N2N\_HO\_Query\_Resources | RP5 |
| MIS\_MN\_HO\_Commit | RP1, RP3 |

Table D.1—Mapping MIS messages to reference points *(continued)*

|  |  |
| --- | --- |
| **MIS message name** | **Reference point** |
| MIS\_Net\_HO\_Commit | RP1, RP3 |
| MIS\_N2N\_HO\_Commit | RP5 |
| MIS\_MN\_HO\_Complete | RP1, RP2, RP3 |
| MIS\_N2N\_HO\_Complete | RP5 |
| MIS\_Get\_Information | RP1, RP2, RP3, RP4, RP5 |
| MIS\_Push\_Information | RP1, RP2, RP3, RP4, RP5 |
| MIS\_Auth | RP1, RP3 |
| MIS\_Termination\_Auth | RP1, RP3 |
| MIS\_Push\_Key | RP1, RP3 |

# (normative) Media specific mapping for SAPs

The MISF aggregates disparate interfaces with respective media dependent lower-layer instances (media dependent service access points) into a single interface with the MIS users (the MIS SAP), reducing the inter-media differences to the extent possible.

The MISF features media dependent interfaces with IEEE 802 link-layer technologies (IEEE 802.2, IEEE 802.3, IEEE 802.11, and IEEE 802.16) and cellular technologies (3GPP and 3GPP2). The MISF for the most part uses existing primitives and functionality provided by different access technology standards. Amendments to existing standards are recommended only when deemed necessary to fulfill the MISF capabilities.

The following subclauses list general amendments recommended to different underlying access technology standards due to the enhanced heterogeneous handover capability provided by MISF.

## MIS\_LINK\_SAP mapping to specific technologies

Table E.1—MIS\_Link\_SAP/IEEE 802.16 primitives mapping

|  |  |  |
| --- | --- | --- |
| **MIS\_LINK\_SAP primitive** | **IEEE Std 802.16 C\_SAP** | **IEEE Std 802.16 M\_SAP** |
| Link\_Detected | C-HO-RSP (HO-Scan) | N/A |
| Link\_Up | C-NEM-RSP (Registration) | N/A |
| Link\_Down | N/A | C-NEM-RSP (Deregistration) |
| Link\_Parameters\_Report | C-HO-IND (HO-Scan) C-HO-RSP (HO-Scan) C-RRM-RSP  C-SFM-RSP | N/A |
| Link\_Going\_Down | N/A | N/A |
| Link\_Handover\_Imminent | C-HO-RSP (HO-Mobile) | N/A |
| Link\_Handover\_Complete | C-NEM-RSP (Ranging) | N/A |
| Link\_PDU\_Transmit\_Status | N/A | N/A |
| Link\_Capability\_Discover | N/A | N/A |
| Link\_Event\_Subscribe | N/A | N/A |
| Link\_Event\_Unsubscribe | N/A | N/A |
| Link\_Get\_Parameters | C-SFM-REQ/RSP C-HO-REQ/RSP/IND (HO-Scan)  C-RRM-REQ/RSP | N/A |
| Link\_Configure\_Thresholds | C-HO-REQ/RSP (HO-Scan) | N/A |

Table E.1—MIS\_Link\_SAP/IEEE 802.16 primitives mapping *(continued)*

|  |  |  |  |
| --- | --- | --- | --- |
| **MIS\_LINK\_SAP primitive** | | **IEEE Std 802.16 C\_SAP** | **IEEE Std 802.16 M\_SAP** |
| Link\_Action | LINK\_DISCONNECT | C-NEM-REQ/RSP (Deregistration) | N/A |
| LINK\_LOW\_POWER | C-IMM-REQ/RSP  (Idle\_Mobile\_Initiation) |  |
| LINK\_POWER\_DOWN | N/A | M-SSM-REQ/RSP (Power down) |
| LINK\_POWER\_UP | N/A | M-SSM-REQ/RSP (Power on) |

Table E.2—MIS\_LINK\_SAP/IEEE 802.11/IEEE 802.3/IEEE 802.1 ag primitives mapping

|  |  |  |  |
| --- | --- | --- | --- |
| **Primitives** | **IEEE Std 802.11** | **IEEE Std 802.3** | **IEEE Std 802.1ag[B19]** |
| Link\_Detected | MSGCF-ESS-Link-Detecteda | N/A | N/A |
| Link\_Up | MSGCF-ESS-Link-Upa | Link fault | dot 1agCfgFaultAlarmb |
| Link\_Down | MSGCF-ESS-Link-Downa | Link fault | dot 1agCfgFaultAlarma |
| Link\_Parameters\_Report | MLME-MEASURE. confirm MLME-MREPORT.indicationc MSGCF-ESS-Link-Threshold­reporta | N/A | N/A |
| Link\_Going\_Down | MSGCF-ESS-Link-Going-Downa | Dying Gasp | N/A |
| Link\_Handover\_Imminent | N/A | N/A | N/A |
| Link\_Handover\_Complete | N/A | N/A | N/A |
| Link\_PDU\_Transmit\_Status | MA-UNIDATA-STATUS.indica- tion | N/A | N/A |
| Link\_Capability\_Discover | N/A | N/A | N/A |
| Link\_Event\_Subscribe | N/A | N/A | N/A |
| Link\_Event\_Unsubscribe | N/A | N/A | N/A |
| Link\_Get\_Parameters | MSGCF-Get-ESS-Link-Parame- tersa | N/A | N/A |
| Link\_Configure\_Thresholds | MLME-MEASURE.request MLME-MREQUEST.requestd MSGCF-Set-ESS-Link-Parame­tersa | N/A | N/A |
| Link\_Action | MSGCF-ESS-Link-Commanda | N/A | N/A |

aSee IEEE P802.11u/3.0

bThe alarms (cross-connection, link failure, MACstatusDefect, and RDIdefect) are enabled and no other higher priority event has occurred.

cIEEE 802.11k MLME-MEASURE.confirm and MLME-MREPORT.indication can be used. If MLME-MEA­SURE.request or MLME-MREQUEST.request includes Beacon Request IE or QoS Metric IE, then MLME-MEA­SURE.confirm or MLME-MREPORT.indication is delivered to the MISF when one of the reporting conditions (thresholds) is satisfied. Link\_Parameter\_Report.indication can be also generated at a predefined regular interval determined by a user configurable time. This is also performed by MLME-MEASURE.request and MLME-MEA­SURE.confirm (local) or MLME-MREQUEST.request and MLME-MREPORT.indication (remote) with measure­ment duration setting.

dIt is used to configure threshold values for Link\_Parameters\_Report. Thresholds are used for triggering reports. IEEE 802.11k primitives, MLME-MEASURE.request(local) and MLME-MREQUEST.request(remote), can be used for that purpose. Only Beacon Request IE and QoS Metric IE can be used for setting thresholds and triggering reports. MLME-MEASURE primitive does not support confirmation to confirm the threshold setting results. It means that MLME-MEASURE primitive does not have the corresponding primitive to Link\_Configure\_Threshold.confirm. MLME-MEASURE.confirm is used to deliver the measurement results not to confirm the threshold setting.

Table E.3—MIS\_LINK\_SAP/3GPP/3GPP2 primitives mapping

|  |  |  |
| --- | --- | --- |
| **Primitives** | **3GPP** | **3GPP2** |
| Link\_Detected | N/A | N/A |
| Link\_Up | SMSM-ACTIVE RABMSM-ACTIVATE | L2.Condition.Notification LCP-Link-Open  LCP-Link-Up  IPCP-Link-Open |
| Link\_Down | SMSM-DEACTIVEATE SMSM-STATUS RABMSM-DEACTIVATE RABMSM-STATUS RABMAS-RAB-RELEASE | LCP-Carrier-Failure LCP-Link-Quality-Failure LCP-Timeout  IPCP-Link-Closed  IPCP-Config-Failure  IPCP-Timeout |
| Link\_Parameters\_Report | SMSM-MODIFY RABMSM-MODIFY | N/A |
| Link\_Going\_Down | N/A | LCP-Closing |
| Link\_Handover\_Imminent | N/A | N/A |
| Link\_Handover\_Complete | RABMAS-RAB-ESTABLISH RABMSM-MODIFY | L2.Data.Confirm |
| Link\_PDU\_Transmit\_Status | N/A | N/A |
| Link\_Capability\_Discover | N/A | N/A |
| Link\_Event\_Subscribe | N/A | N/A |
| Link\_Event\_Unsubscribe | N/A | N/A |
| Link\_Get\_Parameters | N/A | N/A |
| Link\_Configure\_Thresholds | SMREG-PDP-MODIFY | L2.Supervision.Request |
| Link\_Action | N/A | N/A |

## Mappings from MIS\_LINK\_SAP to media-specific SAPs

### IEEE Std 802.3

LSAP, defined in the IEEE Std 802.2, provides the interface between the MISF and the Logical Link Control sublayer in IEEE 802.3 network. This SAP is used for local MIS exchanges between the MISF and the lower layers of the IEEE 802.3 interface (as the IEEE 802.3 instantiation of the MIS\_LINK\_SAP) and for the L2 transport of MIS messages across IEEE 802.3 access links.

### 802.11

The MISF uses MSGCF\_SAP for interfacing with the link layer of IEEE 802.11 networks. The MIS\_LINK\_SAP defines additional primitives that map to MSGCF\_SAP. These primitives are recommended as enhancements to IEEE 802.11 link-layer SAPs. MSGCF\_SAP is defined by IEEE P802.1 1u/D3.0 and it includes, but is not limited to primitives related to the following:

* System configuration
* Link state change notifications/triggers
* MIS frame transport through control or management frames

LSAP, defined in the IEEE Std 802.2, provides the interface between the MISF and the Logical Link Control sublayer in IEEE 802.11. This SAP is used for the L2 transport of MIS messages across IEEE 802.11 access links. The MIS messages are carried in IEEE 802.11 data frames.

Table E.2 lists this mapping.

### IEEE Std 802.16

The MISF uses C\_SAP and M\_SAP for interfacing with the Control and Management planes of the IEEE 802.16 network.

C\_SAP is defined by IEEE Std 802.16gTM-2007 [B21] and it includes primitives related to the following:

* Handovers [e.g., notification of HO request from mobile station (MS)]
* Idle mode mobility management (e.g., Mobile entering idle mode)
* Subscriber and session management (e.g., Mobile requesting session setup)
* Radio resource management
* Authentication, Authorization, and Accounting (AAA) server signaling (e.g., EAP payloads)
* Media independent function services

M\_SAP is defined by IEEE Std 802.16g-2007 [B21] and it includes primitives related to the following:

* System configuration
* Monitoring statistics
* Notifications triggers
* Multi-mode interface management

CS\_SAP, defined in the IEEE Std 802.16, provides the interface between the MISF and the service-specific Convergence Sublayer in IEEE 802.16 networks. This SAP is used for the L2 transport of MIS messages through data frames across IEEE 802.16 access links.

Table E. 1 lists this mapping.

### 3GPP and 3GPP2

This SAP defines MIS\_3GLINK\_SAP interface between the MISF and the different protocol elements of the 3G system.

3GPP and 3GPP2 service primitives for GERAN, UMTS, long term evolution (LTE), cdma2000, cdma2000-HRPD and UMB are used to access MIS services. This is done by establishing a relationship between the 3GPP/3GPP2 primitives and MIS primitives.

Table E.3 lists this mapping. Note that a 3GPP primitive group can be mapped to more than one MIS primitive, as shown in Table E.3.

# (normative) Data type definition

## General

This annex defines data types used in the IEEE 802.21 standard. Any variable-length data type in this speci­fication contains information needed for determining the end of data.

## Basic data types

The data types defined in this subclause are used as the basis for defining any other data types. All basic data types are for general purpose. The “Binary Encoding Rule” column in Table F. 1 describes the encoding rules used when the data types are carried in MIS protocol messages.

Table F.1—Basic data types

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Definition** | **Binary encoding rule** |
| BITMAP(size) | A bitmap of the specified size. Usually used to represent a list of IDs.  Range: Each bit has a value of '0' or '1'. | A BITMAP(N), where N must be a multiple of 8, is made up of an N/8 octet values and encoded in network byte order. |
| CHOICE(  DATATYPE1, DATATYPE2[,... ]) | A data type that consists of only one of the data types listed:  DATATYPE1,DATATYPE2[,... ]. | A one-octet Selector field, followed by a variable length Value field. The Selector value determines the data type. If Selector==i, (i+1)-th data type in the list of data types DATATYPE1,DATATYPE2[,... ] is selected. The Selector value is encoded as UNSIGNED\_INT(1). The Value field is encoded using the encoding rule for the selected data type. |
| INFO\_ELEMENT | A binary encoded structure for Informa- tion Elements. | See 6.5.6. |
| INTEGER(size) | A signed integer of the specified size in number of octets.  Range: Each octet has a value of 0x00 to 0xff. | Each octet of an INTEGER(N) value [N=1,2,...] is encoded in network-byte order into an N-octet field.  The most significant bit of the first octet is the sign bit. If the sign bit is set, it indicates a negative integer. Otherwise, it indicates a non-negative integer.  A negative integer is encoded as 2s complement. |
| LIST(DATATYPE) | A list of values of DATATYPE | See F.2 for details. |
| NULL | A data type with empty data. | No octet is encoded for this data type. This data type is used to define an optional data type. |
| OCTET(size) | An array of octets. The size specifies the length. | The octets are encoded in network byte order. |
| SEQUENCE( DATATYPE1, DATATYPE2[,...]) | A data type that consists of two or more data types. | DATATYPE1,DATATYPE2[,... ] are encoded in the order of appearance. Each data type is encoded using the encoding rule for the data type. |
| UNSIGNED\_INT(size) | An unsigned integer of the specified size in number of octets.  Range: Each octet has a value of 0x00 to 0xff. | Each octet of an UNSIGNED\_INT(N) value [N=1,2,...] is encoded in net­work-byte order into an N-octet field. |

The encoding rule for LIST(DATATYPE) is a variable length *Length* field followed by a variable length *Value* field. The *Length* field shall be interpreted as follows:

Case 1: If the number of list elements in the *Value* field is less than 128, the size of the *Length* field is always one octet and the MSB of the octet is set to the value ‘0’. The values of the other seven bits of this octet indi­cate the actual number of list elements in the *Value* field.

Case 2: If the number of list elements in the *Value* field is exactly 128, the size of the *Length* field is one octet. The MSB of the *Length* octet is set to the value '1' and the other seven bits of this octet are all set to the value ‘0’.

Case 3: If the number of list elements in the *Value* field is greater than 128, then the *Length* field is always greater than one octet. The MSB of the first octet of the *Length* field is set to the value ‘1’ and the remaining seven bits of the first octet indicate the number of octets that are appended further. The number represented by the 2nd and subsequent octets of the *Length* field, when added to 128, indicates the total number of list elements in the *Value* field.

For example, an attribute of type LIST(LINK\_ID) with two elements is encoded as shown in Figure F. 1 (LINK\_ID is defined in F.3.4):

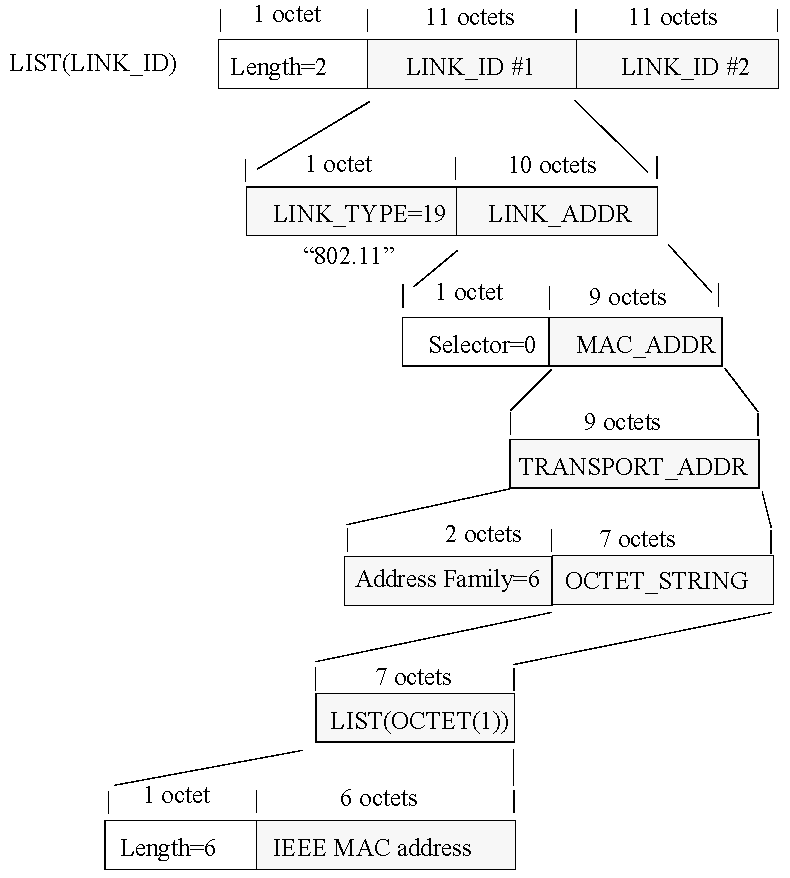


Figure F.1—Encoding example of a LIST with two LINK\_ID elements

## Derived data types

### General

Derived data types are those that are derived from other data types or parent data types. A derived data type uses the same encoding as the parent data type.

### General data types

The derived data types defined in this subclause are for general purpose only.

Table F.2—General data types

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| ENUMERATED | UNSIGNED\_INT(1) | An enumerated attribute. Valid Range: 0..255 |
| BOOLEAN | ENUMERATED | Represents a Boolean.  0: FALSE 1: TRUE |
| OCTET\_STRING | LIST(OCTET(1)) | An array of arbitrary length octets. The default encoding format is UTF-8 [B23]. If a data type derived from OCTET\_STRING uses other encoding format(s), the encoding format(s) must be specified in the definition of such a data type. |
| PERCENTAGE | UNSIGNED\_INT(1) | Represents a percentage. Valid Range: 0..100 |
| STATUS | ENUMERATED | The status of a primitive execution.  0: Success   1. Unspecified Failure 2. Rejected 3. Authorization Failure 4. Network Error 5. Authentication Failure |

### Data types for addresses

The data types defined in this subclause are related to addresses of network elements.

Table F.3—Data types for address

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| 3GPP\_2G\_CELL\_ID | SSEQUENCE(PLMN\_ID, LAC, CI  ) | A data type to represent a 3GPP 2G cell identifier. |
| 3GPP\_3G\_CELL\_ID | SSEQUENCE(PLMN\_ID, CELL\_ID  ) | A data type to represent a 3GPP 3G cell identifier. |
| 3GPP\_ADDR | OCTET\_STRING | A data type to represent a 3GPP transport address. |
| 3GPP2\_ADDR | OCTET\_STRING | A data type to represent a 3GPP2 transport address. |
| CELL\_ID | UNSIGNED\_INT(4) | This data type identifies a cell uniquely within 3GPP UTRAN and consists of radio network controller (RNC)-ID and C-ID as defined in 3GPP TS 25.401.  Valid Range: 0..268435455 |
| CI | OCTET(2) | The BSS and cell within the BSS are identified by Cell Identity (CI). See 3GPP TS 23.003. |
| IP\_ADDR | TRANSPORT\_ADDR | Represents an IP address. The Address Type is either 1 (IPv4) or 2 (IPv6). |
| LAC | OCTET(2) | Location Area Code (LAC) is a fixed length code (of 2 octets) identifying a location area within a public land mobile network (PLMN). See 3GPP TS 23.003. |
| LINK\_ADDR | CHOICE(  MAC\_ADDR, 3GPP\_3G\_CELL\_ID, 3GPP\_2G\_CELL\_ID, 3GPP\_ADDR, 3GPP2\_ADDR, OTHER\_L2\_ADDR  ) | A data type to represent an address of any link layer. |
| MAC\_ADDR | TRANSPORT\_ADDR | Represents a MAC address. The Address Type contains the one used for a specific link layer. |
| OTHER\_L2\_ADDR | OCTET\_STRING | A data type to represent a link-layer address other than the address already defined. For example, SSID. |

Table F.3—Data types for address *(continued)*

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| PLMN\_ID | OCTET(3) | The public land mobile network (PLMN) unique identi­fier. PLMN\_ID consists of Mobile Country Code (MCC) and Mobile Network Code (MNC). This is to represent the access network identifier.  Coding of PLMN\_ID is defined in 3GPP TS 25.4 13. |
| TRANSPORT\_ADDR | SEQUENCE( UNSIGNED\_INT(2), OCTET\_STRING  ) | A type to represent a transport address. The UNSIGNED\_INT(2) is the address type defined in [http://www.iana.org/assignments/address-family-num­bers.](http://www.iana.org/assignments/address-family-num-bers.) |

### Data types for link identification and manipulation

The data types defined in this subclause are used for representing attributes for identification and manipula­tion of links.

Table F.4—Data types for links

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| BATT\_LEVEL | INTEGER(1) | Represents percentage of battery charge remaining.  Valid Range: –1..100.  –1 indicates battery level unknown. |
| CHANNEL\_ID | UNSIGNED\_INT(2) | Channel identifier as defined in the spe­cific link technology (e.g., standards development organization (SDO)).  Valid Range: 0..65535 |
| CONFIG\_STATUS | BOOLEAN | The status of link parameter configura­tion.  TRUE: Success FALSE: Error |
| DEVICE\_INFO | OCTET\_STRING | A non-NULL terminated string whose length shall not exceed 253 octets, rep­resenting information on manufacturer, model number, revision number of the software/firmware and serial number in displayable text. |
| DEV\_STATES\_REQ | BITMAP(16) | A list of device status request.  Bitmap Values:  Bit 0: DEVICE\_INFO  Bit 1: BATT\_LEVEL  Bit 2–15: (Reserved) |
| DEV\_STATES\_RSP | CHOICE(  DEVICE\_INFO, BATT\_LEVEL) | Represents a device status. |
| LINK\_AC\_EX\_TIME | UNSIGNED\_INT(2) | Time (in ms) to elapse before an action needs to be taken. A value of 0 indicates that the action will be taken immedi­ately. Time elapsed will be calculated from the instance the command arrives until the time when the execution of the action is carried out.  Valid Range: 0..65535 |
| LINK\_AC\_RESULT | ENUMERATED | Link action result.  0: Success   1. Failure 2. Refused 3. Incapable |
| LINK\_ACTION | SEQUENCE(  LINK\_AC\_TYPE, LINK\_AC\_ATTR ) | Link action. |

Table F.4—Data types for links *(continued)*

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| LINK\_AC\_ATTR | BITMAP(8) | Link action attribute that can be exe­cuted along with a valid link action. Detail description of each attribute is in Table F.6.  Bitmap Values:  Bit 0: LINK\_SCAN  Bit 1: LINK\_RES\_RETAIN Bit 2: DATA\_FWD\_REQ Bit 3–7: (Reserved) |
| LINK\_ACTION\_REQ | SEQUENCE(  LINK\_ID,  CHOICE(NULL, LINK\_ADDR), LINK\_ACTION,  LINK\_AC\_EX\_TIME  ) | A set of handover action request param­eters. The choice of LINK\_ADDR is to provide PoA address information when the LINK\_ACTION contains the  attribute for DATA\_FWD\_REQ. |
| LINK\_ACTION\_RSP | SEQUENCE(  LINK\_ID,  LINK\_AC\_RESULT,  CHOICE(NULL,  LIST(LINK\_SCAN\_RSP))  ) | A set of link action returned results. |
| LINK\_AC\_TYPE | UNSIGNED\_INT(1) | An action for a link. The meaning of each link action is defined in Table F.5.  0: NONE   1. LINK\_DISCONNECT 2. LINK\_LOW\_POWER 3. LINK\_POWER\_DOWN 4. LINK\_POWER\_UP 5–255: (Reserved) |
| LINK\_CMD\_LIST | BITMAP(32) | A list of link commands.  Bitmap Values: Bit 0: Reserved Bit 1: Link\_Event\_Subscribe  Bit 2: Link\_Event\_Unsubscribe  Bit 3: Link\_Get\_Parameters  Bit 4: Link\_Configure\_Thresholds  Bit 5: Link\_Action Bit 6-31: (Reserved) |
| LINK\_CFG\_PARAM | SEQUENCE(  LINK\_PARAM\_TYPE,  CHOICE(NULL, TIMER\_INTERVAL), TH\_ACTION,  LIST(THRESHOLD)  ) | A link configuration parameter.  TH\_ACTION indicates what action to apply to the listed thresholds.  When “Cancel threshold” is selected and no thresholds are specified, then all currently configured thresholds for the given LINK\_PARAM\_TYPE are can­celled.  When “Cancel threshold” is selected and thresholds are specified only those configured thresholds for the given LINK\_PARAM\_TYPE and whose threshold value match what was speci­fied are cancelled.  With “Set one-shot threshold” the listed thresholds are first set and then each of the threshold is cancelled as soon as it is crossed for the first time. |
| LINK\_CFG\_STATUS | SEQUENCE(  LINK\_PARAM\_TYPE, THRESHOLD,  CONFIG\_STATUS  ) | The status of link parameter configura­tion for each threshold specified in the THRESHOLD. |
| LINK\_DESC\_REQ | BITMAP(16) | A set of link descriptors.  Bitmap Values:  Bit 0: Number of Classes of Service Supported  Bit 1: Number of Queues Supported Bits 2–15: (Reserved) |

Table F.4—Data types for links *(continued)*

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| LINK\_DESC\_RSP | CHOICE(NUM\_COS, NUM\_QUEUE) | Descriptors of a link. |
| LINK\_DATA\_RATE | UNSIGNED\_INT(4) | A type to represent the maximum data rate in kb/s.  Valid Range: 0 – 232–1 |
| LINK\_DN\_REASON | UNSIGNED\_INT(1) | Represents the reason of a link down event.  See Table F.7 for the enumeration values. |
| LINK\_EVENT\_LIST | BITMAP(32) | A list of link events. The specified event is selected if the corresponding bit is set to 1.  Bitmap values:  Bit 0: Link\_Detected  Bit 1: Link\_Up  Bit 2: Link\_Down  Bit 3: Link\_Parameters\_Report  Bit 4: Link\_Going\_Down  Bit 5: Link\_Handover\_Imminent  Bit 6: Link\_Handover\_Complete  Bit 7: Link\_PDU\_Transmit\_Status  Bit 8–31: (Reserved) |
| LINK\_GD\_REASON | UNSIGNED\_INT(1) | Represents the reason of a link going down. See Table F.8 for the enumeration values. |
| LINK\_ID | SEQUENCE( LINK\_TYPE LINK\_ADDR ) | The identifier of a link that is not associ­ated with the peer node. The LINK\_ADDR contains the address of this link. |
| LINK\_MISCAP\_FLAG | BITMAP(8) | Represents if MIS capability is sup­ported or not. If the bit is set, it indicates that the capability is supported.  Bitmap values:  Bit 1: event service (ES) supported  Bit 2: command service (CS) supported  Bit 3: information service (IS) supported Bit 0, 4–7: (Reserved) |
| LINK\_PARAM | SEQUENCE(  LINK\_PARAM\_TYPE, CHOICE(LINK\_PARAM\_VAL,  QOS\_PARAM\_VAL)  ) | Represents a link parameter type and value pair. |
| LINK\_PARAM\_802\_11 | UNSIGNED\_INT(1) | A type to represent a link parameter for IEEE 802.11.  0: RSSI of the beacon channel, as defined in IEEE Std 802.11-2007. (This is applicable only for an MN.)   1. No QoS resource available. The cor­responding LINK\_PARAM\_VAL is BOOLEAN set to TRUE when no QoS resources available. (This applicable when the traffic stream to be transmitted is on an access category configured for mandatory admission control and the request for bandwidth was denied by the available APs in the access network). 2. Multicast packet loss rate.   3–255: (Reserved) |
| LINK\_PARAM\_802\_16 | UNSIGNED\_INT(1) | A type to represent a link parameter for IEEE 802.16.  0–255: (Reserved) |
| LINK\_PARAM\_802\_20 | UNSIGNED\_INT(1) | A type to represent a link parameter for IEEE 802.20.  0–255: (Reserved) |

Table F.4—Data types for links *(continued)*

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| LINK\_PARAM\_802\_22 | UNSIGNED\_INT(1) | A type to represent a link parameter for IEEE 802.22.  0–255: (Reserved) |
| LINK\_PARAM\_C2K | UNSIGNED\_INT(1) | A type to represent a link parameter for CDMA2000.  0: PILOT\_STRENGTH 1–255: (Reserved) |
| LINK\_PARAM\_HRPD | UNSIGNED\_INT(1) | A type to represent a link parameter for CDMA2000 HRPD.  0: PILOT\_STRENGTH 1–255: (Reserved) |
| LINK\_PARAM\_EDGE | UNSIGNED\_INT(1) | A type to represent a link parameter for EDGE.  0-255: (Reserved) |
| LINK\_PARAM\_ETH | UNSIGNED\_INT(1) | A type to represent a link parameter for Ethernet.  0-255: (Reserved) |
| LINK\_PARAM\_GEN | UNSIGNED\_INT(1) | A type to represent a generic link parameter that is applicable to any link type.  0: Data Rate—the parameter value is represented as a DATA\_RATE.   1. Signal Strength—the parameter value is represented as a SIG\_STRENGTH. 2. Signal over interference plus noise ratio (SINR)—the parameter value is represented as an UNSIGNED\_INT(2). 3:Throughput (the number of bits suc­cessfully received divided by the time it took to transmit them over the medium) —the parameter value is represented as an UNSIGNED\_INT(2).   4: Packet Error Rate (representing the ratio between the number of frames received in error and the total number of frames transmitted in a link population of interest)—the parameter value is rep­resented as a PERCENTAGE.  5–255: (Reserved) |
| LINK\_PARAM\_GG | UNSIGNED\_INT(1) | A type to represent a link parameter for GSM and GPRS. See 3GPP TS 25.008.  0: RxQual   1. RsLev 2. Mean BEP 3. StDev BEP   4-255: (Reserved) |
| LINK\_PARAM\_QOS | UNSIGNED\_INT(1) | A type to represent QOS\_LIST parame­ters.  0: Maximum number of differentiable classes of service supported.   1. Minimum packet transfer delay for all CoS, the minimum delay over a class population of interest. 2. Average packet transfer delay for all CoS, the arithmetic mean of the delay over a class population of interest. (See B.3.4) 3. Maximum packet transfer delay for all CoS, the maximum delay over a class population of interest. 4. Packet transfer delay jitter for all CoS, the standard deviation of the delay over a class population of inter­est. (See B.3.5.) 5. Packet loss rate for all CoS, the ratio between the number of frames that are transmitted but not received and the total number of frames transmitted over a class population of interest. (See B.3.2.)   6–255: (Reserved) |

Table F.4—Data types for links *(continued)*

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| LINK\_PARAM\_RPT | SEQUENCE(  LINK\_PARAM,  CHOICE(NULL, THRESHOLD)  ) | Represents a link parameter report. Includes an option of the THRESHOLD that was crossed. If no THRESHOLD is included, then this is a periodic report. |
| LINK\_PARAM\_TYPE | CHOICE(  LINK\_PARAM\_GEN, LINK\_PARAM\_QOS, LINK\_PARAM\_GG, LINK\_PARAM\_EDGE, LINK\_PARAM\_ETH, LINK\_PARAM\_802\_11, LINK\_PARAM\_C2K, LINK\_PARAM\_FDD, LINK\_PARAM\_HRPD, LINK\_PARAM\_802\_16, LINK\_PARAM\_802\_20, LINK\_PARAM\_802\_22 ) | Measurable link parameter for which thresholds are being set. |
| LINK\_PARAM\_VAL | UNSIGNED\_INT(2) | The current value of the parameter. The format of the media-dependent value is defined in the respective media specifi­cation standard and the equivalent num­ber of bits (i.e., first bits) of this data  type is used. In case that there are remaining unused bits in the data type, these are marked as all-zeros (‘0’).  Valid Range: 0..65535 |
| LINK\_PARAM\_FDD | UNSIGNED\_INT(1) | A type to represent a link parameter for UMTS. See 3GPP TS 25.215.  0: CPICH RSCP   1. PCCPCH RSCP 2. UTRA carrier RSSI 3. GSM carrier RSSI 4. CPICH Ec/No 5. Transport channel BLER 6. user equipment (UE) transmitted power   7–255: (Reserved) |
| LINK\_POA\_LIST | SEQUENCE(  LINK\_ID,  LIST(LINK\_ADDR)  ) | A list of PoAs for a particular link. The LIST(LINK\_ADDR) is a list of PoA link addresses and is sorted from most preferred first to least preferred last. |
| LINK\_RES\_STATUS | BOOLEAN | Indicates if a resource is available or not.  TRUE: Available  FALSE: Not available. |
| LINK\_SCAN\_RSP | SEQUENCE(  LINK\_ADDR,  NETWORK\_ID, SIG\_STRENGTH ) | Represents a scan response. The LINK\_ADDR contains the PoA link address. The PoA belongs to the NETWORK\_ID with the given  SIG\_STRENGTH. |
| LINK\_STATES\_REQ | BITMAP(16) | Link states to be requested.  Bit 0: OP\_MODE  Bit 1: CHANNEL\_ID  Bit 2–15: (Reserved) |
| LINK\_STATES\_RSP | CHOICE(OP\_MODE,CHANNEL\_ID) | The operation mode or the channel ID of the link. |
| LINK\_STATUS\_REQ | SEQUENCE(  LINK\_STATES\_REQ, LIST(LINK\_PARAM\_TYPE), LINK\_DESC\_REQ  ) | Represents the possible information to request from a link. |

Table F.4—Data types for links *(continued)*

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| LINK\_STATUS\_RSP | SEQUENCE(  LIST(LINK\_STATES\_RSP), LIST(LINK\_PARAM), LIST(LINK\_DESC\_RSP)  ) | A set of link status parameter values correspond to the  LINK\_STATUS\_REQ. |
| LINK\_TUPLE\_ID | SEQUENCE(  LINK\_ID,  CHOICE(NULL, LINK\_ADDR)  ) | The identifier of a link that is associated with a PoA. The LINK\_ID contains the MN LINK\_ADDR. The optional  LINK\_ADDR contains a link address of PoA. |
| LINK\_TYPE | UNSIGNED\_INT(1) | Represents the link type.a  Number assignments:  0: Reserved   1. Wireless - GSM 2. Wireless - GPRS 3. Wireless - EDGE   15: Ethernet   1. Wireless - Other 2. Wireless - IEEE 802.11 3. Wireless - CDMA2000 4. Wireless - UMTS 5. Wireless - cdma2000-HRPD 6. Wireless - IEEE 802.16 7. Wireless - IEEE 802.20 8. Wireless - IEEE 802.22 |
| NUM\_COS | UNSIGNED\_INT(1) | The maximum number of differentiable classes of service supported.  Valid Range: 0..255 |
| NUM\_QUEUE | UNSIGNED\_INT(1) | The number of transmit queues supported.  Valid Range: 0..255 |
| OP\_MODE | UNSIGNED\_INT(1) | The link power mode.  0: Normal Mode   1. Power Saving Mode 2. Powered Down   3–255: (Reserved) |
| SIG\_STRENGTH | CHOICE(INTEGER(1), PERCENTAGE) | Represents the signal strength in dBm unit or its relative value in an arbitrary percentage scale. |
| TH\_ACTION | ENUMERATED | 0: Set normal threshold   1. Set one-shot threshold 2. Cancel threshold |
| THRESHOLD | SEQUENCE(  THRESHOLD\_VAL, THRESHOLD\_X\_DIR ) | A link threshold. The threshold is con­sidered crossed when the value of the link parameter passes the threshold in the specified direction. |
| THRESHOLD\_VAL | UNSIGNED\_INT(2) | Threshold value. The format of the media-dependent value is defined in the respective media specification standard and the equivalent number of bits (i.e., first bits) of this data type is used. In case that there are remaining unused bits in the data type, these are marked as all- zeros (‘0’).  Valid Range: 0..65535 |
| THRESHOLD\_X\_DIR | UNSIGNED\_INT(1) | The direction the threshold is to be crossed.  0: ABOVE\_THRESHOLD  1: BELOW\_THRESHOLD  2–255: (Reserved) |
| TIMER\_INTERVAL | UNSIGNED\_INT(2) | This timer value (ms) is used to set the interval between periodic reports.  Valid Range: 0..65535 |

aThe values defined are made consistent with RADIUS network access server (NAS)-Port-Type definitions as spec­ified by Internet Assigned Numbers Authority (IANA). (See IETF RFC 2865.)

Table F.5—Link actions

|  |  |
| --- | --- |
| **Action name** | **Description** |
| LINK\_DISCONNECT | Disconnect the link connection directly. |
| LINK\_LOW\_POWER | Cause the link to adjust its battery power level to be low power consumption. |
| LINK\_POWER\_DOWN | Cause the link to power down and turn off the radio. |
| LINK\_POWER\_UP | Cause the link to power up and establish L2 connectivity. For UMTS link type, power up lower layers and establish PDP context. |

Table F.6—Link action attributes

|  |  |
| --- | --- |
| **Action name** | **Description** |
| DATA\_FWD\_REQ | This indication requires the buffered data at the old serving PoA entity to be forwarded to the new target PoA entity in order to avoid data loss. This action can be taken imme­diately after the old serving PoS receives MIS\_N2N\_HO\_Commit response message from the new target PoS, or the old serving PoS receives MIS\_Net\_HO\_Commit response message from the MN. This is not valid on UMTS link type. |
| LINK\_RES\_RETAIN | The link will be disconnected but the resource for the link connection still remains so reestablishing the link connection later can be more efficient. |
| LINK\_SCAN | Cause the link to perform a scan. |

Table F.7—Link down reason code

|  |  |  |
| --- | --- | --- |
| **Reason code** | **Reason** | **Description** |
| 0 | Explicit disconnect | The link is down because of explicit disconnect procedures initiated either by MN or network. |
| 1 | Packet timeout | The link is down because no acknowledgements were received for transmitted packets within the specified time limit. |
| 2 | No resource | The link is down because there were no resources to maintain the connection. |
| 3 | No broadcast | The link is down because broadcast messages (such as beacons in IEEE 802.11 management frames) could not be received by MN. |
| 4 | Authentication failure | Authentication failure. |
| 5 | Billing failure | Billing failure. |
| 6–127 | (*Reserved*) | Reserved for IEEE 802.21 future use. |
| 128–255 | Vendor specific rea- son codes | Vendors specify their own specific reason codes in this range. |

Table F.8—Link going down reason code

|  |  |  |
| --- | --- | --- |
| **Reason code** | **Reason** | **Description** |
| 0 | Explicit disconnect | The link is going to be down because explicit disconnect procedures will be initiated either by MN or network. For example, when a BS has decided to shutdown for administrative reasons or an operator of the terminal has decided to execute a handover manually, a Link\_Going\_Down trigger is sent to the MISF. |
| 1 | Link parameter degrading | The link is going to be down because broadcast messages (such as beacons in IEEE 802.11 management frames) could not be received by MN. |
| 2 | Low power | The link is going to be down because the power level of the terminal is low and the current link will not be maintained in such a low power level. Mobile terminals usually have limited battery supply, and when the battery level of the terminal is low, a terminal can choose a link that has lower power con­sumption for handover according to the received Link\_Going\_Down triggers with this reason code. This will lengthen the usable time for the terminal. |
| 3 | No resource | The link is going to be down because there will be no resources to maintain the current connection. For example, a BS that has too many users can send Link\_Going\_Down indications to terminals when the links with them can not be kept because of insufficient resources. Another example is that users with higher priority can preempt the ones with lower priority when no more resources can be allocated in 3GPP, and this can also cause a Link\_Going\_Down indication with this reason code. |
| 4–127 | (*Reserved*) | Reserved for IEEE 802.21 future use. |
| 128–255 | Vendor specific rea- son codes | Vendors specify their own specific reason codes in this range. |

### Data types for QoS

The data types defined in this subclause are related to QoS.

Table F.9—Data types for QoS

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| QOS\_LIST | SEQUENCE(  NUM\_COS\_TYPES, LIST(MIN\_PK\_TX\_DELAY), LIST(AVG\_PK\_TX\_DELAY), LIST(MAX\_PK\_TX\_DELAY), LIST(PK\_DELAY\_JITTER), LIST(PK\_LOSS\_RATE)  ) | A list of Class of Service (CoS) parameters. |
| NUM\_COS\_TYPES | UNSIGNED\_INT(1) | A type to represent the maximum number of differ­entiable classes of service supported.  Valid Range: 0..255 |
| MIN\_PK\_TX\_DELAY | SEQUENCE(  COS\_ID,  UNSIGNED\_INT(2)  ) | A type to represent the minimum packet transfer delay in ms for the specific CoS specified by the COS\_ID. |
| AVG\_PK\_TX\_DELAY | SEQUENCE(  COS\_ID,  UNSIGNED\_INT(2)  ) | A type to represent the average packet transfer delay in ms for the specific CoS specified by the COS\_ID. |
| MAX\_PK\_TX\_DELAY | SEQUENCE(  COS\_ID,  UNSIGNED\_INT(2)  ) | A type to represent the maximum packet transfer delay in ms for the specific CoS specified by the COS\_ID. |
| PK\_DELAY\_JITTER | SEQUENCE(  COS\_ID,  UNSIGNED\_INT(2)  ) | A type to represent the packet transfer delay jitter in ms for the specific CoS specified by the COS\_ID. |
| PK\_LOSS\_RATE | SEQUENCE(  COS\_ID,  UNSIGNED\_INT(2)  ) | A type to represent the packet loss rate for the spe­cific CoS specified by the COS\_ID. The loss rate is equal to the integer part of the result of multiplying –100 times the log10 of the ratio between the num­ber of packets lost and the total number of packets transmitted in the class population of interest. |
| COS\_ID | UNSIGNED\_INT(1) | A type to represent a class of service identifier. Valid Range: 0–255 |
| QOS\_PARAM\_VAL | CHOICE(  NUM\_COS\_TYPES, LIST(MIN\_PK\_TX\_DELAY), LIST(AVG\_PK\_TX\_DELAY), LIST(MAX\_PK\_TX\_DELAY), LIST(PK\_DELAY\_JITTER), LIST(PK\_LOSS\_RATE)  ) | A choice of Class of Service (CoS) parameters. |

### Data types for location

Table F.10—Data types for location

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| LOCATION | CHOICE(  CIVIC\_LOC, GEO\_LOC, CELL\_ID  ) | A type to represent the format and value of the location information. The location can be civic location, geospacial location, or a cellular ID value as reference location. |
| CIVIC\_LOC | CHOICE( BIN\_CIVIC\_LOC, XML\_CIVIC\_LOC ) | A type to represent a civic address. |
| BIN\_CIVIC\_LOC | SEQUENCE( CNTRY\_CODE, CIVIC\_ADDR  ) | A type to represent a binary-formatted civic address. See CNTRY\_CODE and CIVIC\_ADDR definitions. |
| XML\_CIVIC\_LOC | OCTET\_STRING | A type to represent an XML-formatted civic location. Civic address elements, as described in IETF RFC 4119. |
| CIVIC\_ADDR | OCTET\_STRING | A type to represent civic address elements in BIN\_CIVIC\_LOC.  Civic address elements, as described in IETF RFC 4776. |
| GEO\_LOC | CHOICE(  BIN\_GEO\_LOC, XML\_GEO\_LOC ) | A type to represent a geospatial location. |
| BIN\_GEO\_LOC | OCTET(16) | A type to represent a binary-formatted geospatial location. See Table F.11. |
| XML\_GEO\_LOC | OCTET\_STRING | A type to represent an XML-formatted geospatial location. Geo address elements as described in IETF RFC 4119. For example, <gml:location>  <gml:Point gml:id=“point1” srsName=“epsg:4326”>  <gml:coordinates>37:46:30N 122:25: 10W</gml:coordinates>  </gml:Point> </gml:location> |

Table F.11—Value field format of PoA location information (geospatial location )

|  |  |  |
| --- | --- | --- |
| **Syntax** | **Length (bits)** | **Notes (See IETF RFC 3825 for details)** |
| LatitudeResolution (LaRes) | 6 | Latitude resolution: six bits indicating the number of valid bits in the fixed­point value of Latitude. Any bits entered to the right of this limit should not be considered valid and might be purposely false, or zeroed by the sender. |
| Latitude | 34 | A 34-bit fixed point value consisting of nine bits of integer and 25 bits of fraction. Latitude should be normalized to within ± 90 degrees. Positive num­bers are north of the equator and negative numbers are south of the equator. |
| LongitudeResolution (LoRes) | 6 | Longitude resolution: six bits indicating the number of valid bits in the fixed­point value of Longitude. This value is the number of high-order Longitude bits that should be considered valid. Any bits entered to the right of this limit should not be considered valid and might be purposely false, or zeroed by the sender. |
| Longitude | 34 | A 34 bit fixed point value consisting of nine bits of integer and 25 bits of fraction. Longitude should be normalized to within ± 180 degrees. Positive values are East of the prime meridian and negative (2s complement) numbers are West of the prime meridian. |
| AltitudeType (AT) | 4 | Following codes are defined:   1. Meters: in 2s-complement fixed-point 22-bit integer part with 8-bit frac­tion. If AT = 1, an AltRes value 0.0 would indicate unknown altitude. The most precise Altitude would have an AltRes value of 30. Many values of AltRes would obscure any variation due to vertical datum differences. 2. Floors: in 2s-complement fixed-point 22-bit integer part with 8-bit frac­tion. AT = 2 for Floors enables representing altitude in a form more relevant in buildings that have different floor-to-floor dimensions. |
| AltitudeResolution (AltRes) | 6 | Altitude resolution: six bits indicating the number of valid bits in the altitude. Values above 30 (decimal) are undefined and reserved. |
| Altitude | 30 | A 30-bit value defined by the AT field. |
| Datum | 8 | Following codes are defined:   1. WGS 2. NAD 83 (with associated vertical datum for North American vertical datum for 1998) 3. NAD 83 [with associated vertical datum for Mean Lower Low Water (MLLW)] |

### Data types for IP configuration

Table F.12—Data types for IP configuration

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| IP\_CFG\_MTHDS | BITMAP(32) | A set of IP configuration methods.  Bit 0: IPv4 static configuration  Bit 1: IPv4 dynamic configuration (DHCPv4)  Bit 2: Mobile IPv4 with foreign agent (FA) care-of address (CoA) (FA-CoA)  Bit 3: Mobile IPv4 without FA (Co-located CoA) Bits 4–10: reserved for IPv4 address configurations Bit 11: IPv6 stateless address configuration  Bit 12: IPv6 stateful address configuration (DHCPv6) Bit 13: IPv6 manual configuration  Bits 14–31: (Reserved) |
| IP\_MOB\_MGMT | BITMAP(16) | Indicates the supported mobility management protocols.  Bit 0: Mobile IPv4 (IETF RFC 3344)  Bit 1: Mobile IPv4 Regional Registration (IETF RFC 4857)  Bit 2: Mobile IPv6 (IETF RFC 3775)  Bit 3: Hierarchical Mobile IPv6 (IETF RFC 4140)  Bit 4: Low Latency Handoffs (IETF RFC 4881)  Bit 5: Mobile IPv6 Fast Handovers (IETF RFC 5268)  Bit 6: IKEv2 Mobility and Multihoming Protocol (IETF RFC 4555)  Bit 7–15: (Reserved) |
| IP\_PREFIX\_LEN | UNSIGNED\_INT(1) | The length of an IP subnet prefix.  Valid Range:  0..32 for IPv4 subnet.  0..64, 65.. 127 for IPv6 subnet. (IETF RFC 4291 [B25]) |
| IP\_RENEWAL\_FLAG | BOOLEAN | Indicates whether MN’s IP address needs to be changed or not.  TRUE: Change required. FALSE: Change not required. |
| IP\_SUBNET\_INFO | SEQUENCE(  IP\_PREFIX\_LEN, IP\_ADDR  ) | Represent an IP subnet. The IP\_PREFIX\_LEN contains the bit length of the prefix of the subnet to which the IP\_ADDR belongs. |

### Data types for information elements

Data types defined in this subclause are used only by IEs.

Table F.13—Data types for information elements

| **Data type name** | **Derived from** | **Definition** |
| --- | --- | --- |
| NET\_AUX\_ID | OCTET\_STRING | A type to represent an auxiliary access network identifier. This is HESSID if network type is IEEE 802.11. |
| NETWORK\_ID | OCTET\_STRING | A type to represent a network identifier.  A non-NULL terminated string whose length shall not exceed 253 octets. |
| BAND\_CLASS | UNSIGNED\_INT(1) | CDMA band class. |
| BANDWIDTH | UNSIGNED\_INT(2) | Channel bandwidth in kb/s. |
| BASE\_ID | UNSIGNED\_INT(2) | Base station identifier. |
| BURST\_PROF | SEQUENCE( DOWN\_BP, UP\_BP  ) | Burst profile |
| CH\_RANGE | SEQUENCE(  UNSIGNED\_INT(4), UNSIGNED\_INT(4) ) | A type that contains two numbers. The first unsigned integer is the low range. The second unsigned integer is the high range. Both values are in kHz.  The first unsigned integer value should always be less than or equal to the second unsigned integer. |
| COST | SEQUENCE(  COST\_UNIT, COST\_VALUE, COST\_CURR ) | A type to represent a cost. |
| COST\_CURR | OCTET(3) | A type to represent the currency of a cost.  A three-letter currency code (e.g., “USD”) specified by ISO 4217. |
| COST\_UNIT | UNSIGNED\_INT(1) | A type to represent the unit of a cost.  0: second   1. minute 2. hours 3. day 4. week 5. month 6. year 7. free 8. flat rate 9–255: (Reserved) |
| COST\_VALUE | SEQUENCE(  UNSIGNED\_INT(4), UNSIGNED\_INT(2) ) | A type to represent the value of a cost.  The first 4-octet contains the integer part of the cost. The last 2-octet contains the fraction part where it rep­resents a 3-digit fraction.  Therefore, the value range of the fraction part is [0,999].  For example, for a value of “0.5”, the integer part is zero and the fraction part is 500. |
| CNTRY\_CODE | OCTET(2) | Country code, represented as two letter ISO 3 166-1 country code in capital ASCII letters. |
| DATA\_RATE | UNSIGNED\_INT(4) | A type to represent the maximum data rate in kb/s. Valid Range: 0..232–1 |
| DCD\_UCD | SEQUENCE(  BASE\_ID,  BANDWIDTH, DU\_CTR\_FREQ, EIRP,  GAP,  BURST\_PROF, CDMA\_CODES  ) | A type to represent the downlink channel descriptor and the uplink channel descriptor. |
| DOWN\_BP | BITMAP(256) | A List of FEC Code Type for Downlink burst. Refer to 11.4.1 in IEEE 802.16Rev2/D5.0. |
| EIRP | INTEGER(1) | BS’s effective isotropic radiated power level. Signed in units of 1 dBm. |
| FQDN | OCTET\_STRING | The fully qualified domain name of a host as described in IETF RFC 2181. |
| DU\_CTR\_FREQ | INTEGER(8) | Downlink/Uplink center frequency in kHz. |
| FREQ\_BANDS | LIST(UNSIGNED\_INT(4)) | A list of frequency bands. The values are in kHz. |
| FREQ\_ID | INTEGER(2) | Identifier of the carrier frequency. Valid Range: 0..65535 |
| FQ\_CODE\_NUM | INTEGER(2) | UMTS scrambling code, cdma2000 Walsh code. Valid Range: 0..65535 |
| GAP | SEQUENCE(  UNSIGNED\_INT(2), UNSIGNED\_INT(1) ) | This gap is an integer number of physical slot dura­tions and starts on a physical slot boundary. Used on TDD systems only.  The UNSIGNED\_INT(2) is used for the TTG - trans­mit/receive transition gap.  The UNSIGNED\_INT(1) is used for the RTG - receive/transmit transition gap. |
| HO\_CODE | INTEGER(1) | HANDOVER\_RANGING\_CODE.  Refer to 11.3.1 in IEEE 802. 16Rev2/D5.0. |
| INIT\_CODE | INTEGER(1) | INITIAL\_RANGING\_CODE.  Refer to 11.3.1 in IEEE 802. 16Rev2/D5.0. |
| IP4\_ADDR | OCTET(4) | An IPv4 address as described in IETF RFC 791 [B22]. |
| IP6\_ADDR | OCTET(16) | An IPv6 address as described in IETF RFC 4291 [B25]. |
| IP\_CONFIG | SEQUENCE(  IP\_CFG\_MTHDS, CHOICE(NULL, DHCP\_SERV), CHOICE(NULL, FN\_AGNT), CHOICE(NULL, ACC\_RTR)  ) | IP Configuration Methods supported by the access network. |
| NET\_CAPS | BITMAP(32) | These bits provide high level capabilities supported on a network.  Bitmap Values:  Bit 0: Security – Indicates that some level of security is supported when set.  Bit 1: QoS Class 0 – Indicates that QoS for class 0 is supported when set.a  Bit 2: QoS Class 1 – Indicates that QoS for class 1 is supported when set.a  Bit 3: QoS Class 2 – Indicates that QoS for class 2 is supported when set; Otherwise, no QoS for class 2 support is available.  Bit 4: QoS Class 3 – Indicates that QoS for class 3 is supported when set; Otherwise, no QoS for class 3 support is available.  Bit 5: QoS Class 4 – Indicates that QoS for class 4 is supported when set; Otherwise, no QoS for class 4 support is available.  Bit 6: QoS Class 5 – Indicates that QoS for class 5 is supported when set; Otherwise, no QoS for class 5 support is available.  Bit 7: Internet Access – Indicates that Internet access is supported when set; Otherwise, no Internet access support is available.  Bit 8: Emergency Services – Indicates that some level of emergency services is supported when set; Other­wise, no emergency service support is available.  Bit 9: MIS Capability – Indicates that MIS is sup­ported when set; Otherwise, no MIS support is avail­able.  Bit 10–31: (Reserved) |
| NETWORK\_TYPE | SEQUENCE(  CHOICE(NULL, LINK\_TYPE), CHOICE(NULL, SUBTYPE), CHOICE(NULL, TYPE\_EXT) ) | A type to represent a network type and its subtype. See Table F.14 for details. |
| OPERATOR\_ID | SEQUENCE(  OP\_NAME,  OP\_NAMESPACE  ) | A type to represent an operator identifier. |
| OP\_NAME | OCTET\_STRING | A type to represent an operator name. The value uniquely identifies the operator name within the scope of the OP\_NAMESPACE.  The value is a non NULL terminated string whose length shall not exceed 253 octets. |
| OP\_NAMESPACE | UNSIGNED\_INT(1) | A type to represent a type of operator name.  0: GSM/UMTS   1. CDMA 2. REALM (as defined in [B28]). 3. ITU-T/TSB 4. General   5–255: (Reserved) |
| PARAMETERS | CHOICE(  DCD\_UCD,  SIB,  SYS\_PARAMS  ) | A data type to represent system information depend­ing on the network type.  DCD\_UCD: IEEE 802.16  SIB: UMTS  SYS\_PARAMS: cdma2000 |
| PILOT\_PN | INTEGER(2) | Pilot PN sequence offset index. |
| PROXY\_ADDR | CHOICE(  IP4\_ADDR, IP6\_ADDR, FQDN  ) | L3 address of a proxy server. |
| CDMA\_CODES | SEQUENCE(INIT\_CODE, HO\_CODE) | A set of CDMA ranging codes. |
| REGU\_DOMAIN | SEQUENCE(  CNTRY\_CODE,  UNSIGNED\_INT(1)  ) | A type to represent a regulatory domain. A regulatory domain is identified by a country code (CNTRY\_CODE) and a regulatory class  (UNSIGNED\_INT(1)). The regulatory class values are defined in Annex J of IEEE Std 802.11k |
| SUBTYPE | BITMAP(64) | A network subtype. See Table F.14. |
| ROAMING\_PTNS | LIST(OPERATOR\_ID) | A list of roaming partners. |
| SP\_ID | OCTET\_STRING | A service provider identifier.  A non-NULL terminated string whose length shall not exceed 253 octets. |
| SIB | SEQUENCE(  CELL\_ID,  FQ\_CODE\_NUM  ) | A type to represent UMTS system information block (SIB). |

Table F.13—Data types for information elements *(continued)*

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| SUPPORTED\_LCP | UNSIGNED\_INT(1) | A type represent supported Location Configuration Protocol. (LCP). Location by Reference (LbyR).  Values represent LCPs: 0: NULL   1. LLDP 2. LbyR with LLDP 3–10: (Reserved) 3. LLDP-MED 4. LbyR with LLDP-MED   13–20: (Reserved)   1. U-TDoA 2. D-TDoA   23–30: (Reserved)   1. DHCP 2. LbyR with DHCP 33–40: (Reserved) 3. OMA SUPL 4. IEEE 802.11 5. LbyR with IEEE 802.11   44–50: (Reserved)   1. HELD 2. LbyR with HELD 53–255: (Reserved) |
| SYSTEM\_INFO | SEQUENCE(  NETWORK\_TYPE, LINK\_ADDR,  CHOICE(NULL, PARAMETERS)  ) | A type to represent system information. |
| SYS\_PARAMS | SEQUENCE ( BASE\_ID, PILOT\_PN, FREQ\_ID, BAND\_CLASS  ) | CDMA2000 system parameters. |
| TYPE\_EXT | OCTET\_STRING | A generic type extension contained indicating a flexi­ble length and format field. The content is to be defined and filled by the appropriate SDO or service provider consortium, etc.  The value is a non-NULL terminated string whose length shall not exceed 253 octets. |
| UP\_BP | BITMAP(256) | A List of FEC Code Type for Uplink burst. Refer to 11.3.1 in IEEE 802. 16Rev2/D5.0 |

aThe definitions of the QOS classes are according to ITU Y. 1541 [B 34].

Table F.14—Network type and subtype representation

|  |  |  |
| --- | --- | --- |
| **Network** | **Link type** | **Network subtype** |
| (*Reserved*) | 0 | N/A |
| Wireless - GSM | 1 | N/A |
| Wireless - GPRS | 2 | N/A |
| Wireless - EDGE | 3 | N/A |
| (*Reserved*) | 4-14 | N/A |
| Ethernet - IEEE 802.3 | 15 | Bit 0: 10 Mb  Bit 1: 100 Mb  Bit 2: 1000 Mb  Bit 3–63: (Reserved)  The above bits represent the link speeds that Ethernet supports. The capability information of twisted pair Ethernet link can be obtained via auto-negotiation as defined in Clause 28 of IEEE Std 802.3. |
| (*Reserved*) | 16-17 | N/A |
| Wireless - Other | 18 | N/A |
| Wireless - IEEE 802.11 | 19 | Bit 0: 2.4 GHz Bit 1: 5 GHz Bit 2: 4.9 GHz Bit 3: 3.65 GHz Bit 4: 316 THz Bit 5-63 (Reserved)  The above bits represent the frequency band that IEEE 802.11  link supports. The capability information and extended capabili­ties information of IEEE 802.11 link can further be represented  as defined in 7.3.1.4 and 7.3.2.27, respectively, of IEEE Std  802. 11-2007. |
| (*Reserved*) | 20-2 1 | N/A |
| Wireless - CDMA2000 | 22 | N/A |
| Wireless - UMTS | 23 | Bit 0: Rel-99  Bit 1: Rel-4  Bit 2: Rel-5 (w/ HSDPA) Bit 3: Rel-6 (w/ HSUPA)  Bit 4: Rel-7 (MIMO/OFDM) Bit 5: Rel-8  Bit 6–63: (Reserved) |
| Wireless - cdma2000-HRPD | 24 | Bit 0: Rev-0 Bit 1: Rev-A Bit 2: Rev-B Bit 3: Rev-C Bit 4–63: (Reserved) |
| (*Reserved*) | 25-26 | N/A |
| Wireless - IEEE 802.16 | 27 | Bit 0: 2.5 GHz  Bit 1: 3.5 GHz  Bit 2-63: (Reserved)  The above bits represent the frequency band that IEEE 802.16 link supports. The system profiles of IEEE 802.16 link can fur­ther be represented as defined in clause 12 (12.3 and 12.4) of IEEE Std 802. 16e-2005. |
| Wireless - IEEE 802.20 | 28 | N/A |
| Wireless - IEEE 802.22 | 29 | N/A |
| (*Reserved*) | 30-255 | N/A |

NOTE—The Link type values in Table F.14 are deliberately made consistent with RADIUS network access server (NAS)-Port-Type definitions as specified by Internet Assigned Numbers Authority (IANA).

### Data types for information service query

#### Binary representation

Table F.15—Data types for binary query

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| CURR\_PREF | COST\_CURR | A type to indicate currency preference. |
| IE\_TYPE | UNSIGNED\_INT(4) | A type to represent an IE type. See Table G. 1 for more information. |
| IQ\_BIN\_DATA | SEQUENCE(  CHOICE(NULL, QUERIER\_LOC), CHOICE(NULL, NET\_TYPE\_INC), CHOICE(NULL, NETWK\_INC), CHOICE(NULL, RPT\_TEMPL), CHOICE(NULL, RPT\_LIMIT), CHOICE(NULL, CURR\_PREF)  ) | Represents a binary query.  There should exist at least one of the query data type QUERIER\_LOC, NET\_TYPE\_INC, or NETWK\_INC.  One CURR\_PREF at most is included in an Info Query Binary TLV. If included, it indicates to the MIIS server the preferred currency the returned cost should be represented in. If the MIIS server cannot return the cost in the specified currency, it can return the cost in other currencies. |
| NGHB\_RADIUS | UNSIGNED\_INT(4) | The radius in meters from the center point of querier’s location.  Valid Range: 0..232–1 |
| NETWK\_INC | LIST(NETWORK\_ID) | A type to represent a list of network identifiers. |

Table F.15—Data types for binary query (continued)

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| NET\_TYPE\_INC | BITMAP(32) | A type to represent a set of link types.  The value is a four octet bitmap:  Bit 0: Wireless - GSM  Bit 1: Wireless - GPRS  Bit 2: Wireless - EDGE  Bit 3: IEEE 802.3 (Ethernet)  Bit 4: Wireless - Other  Bit 5: Wireless - IEEE 802.11  Bit 6: Wireless - CDMA2000  Bit 7: Wireless - UMTS  Bit 8: Wireless - cdma2000-HRPD  Bit 9: Wireless - IEEE 802.16  Bit 10: Wireless - IEEE 802.20  Bit 11: Wireless - IEEE 802.22  Bit 12–31: (*Reserved AND shall be always set to “0”*) |
| QUERIER\_LOC | SEQUENCE(  CHOICE(NULL, LOCATION), CHOICE(NULL, LINK\_ADDR), CHOICE(NULL, NGHB\_RADIUS)  ) | A type to represent a querier's location. It is not valid to use both LOCATION and LINK\_ADDR at the same time. |
| RPT\_LIMIT | SEQUENCE(  UNSIGNED\_INT(2), UNSIGNED\_INT(2)  ) | A type to represent a report limitation. The first UNSIGNED\_INT(2) contains the maxi­mum number of IEs in the IR\_BIN\_DATA. The second UNSIGNED\_INT(2) contains the starting entry number (offset = 1 points to the first entry) from which a chunk of entries are to be included in the IQ\_BIN\_DATA. It is assumed that the IS server generates the same ordered list of entries for queries from the same IS client with the same IR\_BIN\_DATA content (except for RPT\_LIMIT) before the limitation on the RPT\_LIMIT is applied. |
| RPT\_TEMPL | LIST(IE\_TYPE) | A type to represent a list of IE types. Inclusion of any IE type is optional. |

#### RDF representation

Table F.16—Data type for RDF query

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| IQ\_RDF\_SCHM | OCTET\_STRING | A type to represent the URL of an RDF schema to obtain. |
| IQ\_RDF\_DATA | SEQUENCE(  CHOICE(NULL, MIME\_TYPE), OCTET\_STRING  ) | Represents RDF query. If MIME\_TYPE is oMISed, MIME type “application/sparql-query” is used. Each OCTET\_STRING is formatted with the MIME type. |
| MIME\_TYPE | OCTET\_STRING | Represents MIME type. |

### Data types for information service response

#### Binary representation

Table F.17—Data type for binary information query response

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| IR\_BIN\_DATA | LIST(INFO\_ELEMENT) | A type to represent a binary query response data. |

#### RDF representation

Table F.18—Data type for RDF information query response

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| IR\_RDF\_DATA | SEQUENCE(  CHOICE(NULL, MIME\_TYPE), OCTET\_STRING  ) | Represents RDF data query result. If MIME\_TYPE is oMISed, MIME type “application/ sparql‑  results+xml” is used. OCTET\_STRING is formatted with the MIME type. |
| IR\_SCHM\_URL | OCTET\_STRING | An URL of an RDF schema. |
| IR\_RDF\_SCHM | SEQUENCE(  CHOICE(NULL, MIME\_TYPE), OCTET\_STRING  ) | Represents an RDF schema. If MIME\_TYPE is oMISed, MIME type “application/xml” is used. OCTET\_STRING is formatted with the MIME type. |

### Data type for MISF identification

Table F.19—Data type for MISF identification

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| MISF\_ID | OCTET\_STRING | The MISF Identifier: MISF\_ID is a network access identifier (NAI). |
|  |  | NAI shall be unique as per IETF RFC 4282. If L3 communication is used and MISF entity resides in the network node, then MISF\_ID is the fully qualified domain name or NAI-encoded IP address |
|  |  | (IP4\_ADDR or IP6\_ADDR) of the entity that hosts the MIS Services. |
|  |  | If L2 communication is used then MISF\_ID is the NAI-encoded link- layer address (LINK\_ADDR) of the entity that hosts the MIS ser­vices. In an NAI-encoded IP address or link-layer address, each octet of binary-encoded IP4\_ADDR, IP6\_ADDR and LINK\_ADDR data is encoded in the username part of the NAI as “\” followed by the octet value. A multicast MISF identifier is defined as an MISF ID of zero length. When an MIS protocol message with multicast MISF ID is transmitted over the L2 data plane, a group MAC address (01-80-C2- |
|  |  | 00-00-0E) shall be used (see IEEE P802.1aj/D2.2). The maximum length is 253 octets. |

### Data type for MIS capabilities

Table F.20—Data type for MIS capabilities

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| EVT\_CFG\_INFO | CHOICE(  LIST(LINK\_DET\_CFG), LIST(LINK\_CFG\_PARAM)  ) | Represents additional configuration information for event subscription. The list of LINK\_DET\_CFG contains additional filtering when subscribing to link detected events. The list of LINK\_CFG\_PARAM contains additional fil­tering when subscribing to link parameter report events. |
| LINK\_DET\_CFG | SEQUENCE(  CHOICE(NULL, NETWORK\_ID), CHOICE(NULL,  SIG\_STRENGTH), CHOICE(NULL, LINK\_DATA\_RATE)  ) | A data type for configuring link detected event trigger. |
| LINK\_DET\_INFO | SEQUENCE (  LINK\_TUPLE\_ID, NETWORK\_ID,  NET\_AUX\_ID,  SIG\_STRENGTH, UNSIGNED\_INT(2), LINK\_DATA\_RATE, LINK\_MISCAP\_FLAG, NET\_CAPS  ) | Information of a detected link.  LINK\_TUPLE\_ID is the link detected. NETWORK\_ID is the access network identifier. NET\_AUX\_ID is an auxiliary access network identifier if applicable.  SIG\_STRENGTH is the signal strength of the detected link.  UNSIGNED\_INT(2) is the SINR value of the link.  LINK\_DATA\_RATE is the maximum transmis­sion rate on the detected link.  LINK\_MISCAP\_FLAG indicates which MIS capabilities are supported on the detected link. NET\_CAPS is the network capability supported by the network link. |
| MBB\_HO\_SUPP | SEQUENCE(  NETWORK\_TYPE, NETWORK\_TYPE, BOOLEAN  ) | Indicates if make before break is supported FROM the first network type TO the second net­work type.  The BOOLEAN value assignment:  True: Make before break is supported. False: Make before break is not supported. |
| MIS\_CMD\_LIST | BITMAP(32) | A list of MIS commands.  Bitmap Values:  Bit 0: MIS\_Link\_Get\_Parameters  Bit 1: MIS\_Link\_Configure\_Thresholds Bit 2: MIS\_Link\_Actions  Bit 3: MIS\_Net\_HO\_Candidate\_Query MIS\_Net\_HO\_Commit MIS\_N2N\_HO\_Query\_Resources MIS\_N2N\_HO\_Commit MIS\_N2N\_HO\_Complete  Bit 4: MIS\_MN\_HO\_Candidate\_Query MIS\_MN\_HO\_Commit MIS\_MN\_HO\_Complete  Bit 29: IE\_AUTHENTICATOR\_LINK\_ADDR  Bit 30: IE\_AUTHENTICATOR\_IP\_ADDR  Bit 31: IE\_PoS\_IP\_ADDR  Bit 32: IE\_KEY\_DIST\_INF  Bit 33: IE\_PoS\_INTG\_ALG\_INF  Bit 34: IE\_PoS\_ENCR\_ALG\_INF  Bit 35: IE\_PoS\_PRF\_INF  Bit 36– 63 (Reserved) |

**Table F.20—Data type for MIS capabilities *(continued)***

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| MIS\_EVT\_LIST | BITMAP(32) | A list of MIS events. |
|  |  | Bitmap Values: |
|  |  | Bit 0: MIS\_Link\_Detected |
|  |  | Bit 1: MIS\_Link\_Up |
|  |  | Bit 2: MIS Link\_Down |
|  |  | Bit 3: MIS\_Link\_Parameters\_Report |
|  |  | Bit 4: MIS\_Link\_Going\_Down |
|  |  | Bit 5: MIS\_Link\_Handover\_Imminent |
|  |  | Bit 6: MIS\_Link\_Handover\_Complete |
|  |  | Bit 7: MIS\_Link\_PDU\_Transmit\_Status  Bit 8–31: (Reserved)  A list of IS query types. |
| MIS\_IQ\_TYPE\_LST | BITMAP(64) | Bitmap Values: |
|  |  | Bit 0: Binary data |
|  |  | Bit 1: RDF data |
|  |  | Bit 2: RDF schema URL |
|  |  | Bit 3: RDF schema |
|  |  | Bit 4: IE\_NETWORK\_TYPE |
|  |  | Bit 5: IE\_OPERATOR\_ID |
|  |  | Bit 6: IE\_SERVICE\_PROVIDER\_ID |
|  |  | Bit 7: IE\_COUNTRY\_CODE |
|  |  | Bit 8: IE\_NETWORK\_ID |
|  |  | Bit 9: IE\_NETWORK\_AUX\_ID |
|  |  | Bit 10: IE\_ROAMING\_PARTNERS |
|  |  | Bit 11: IE\_COST |
|  |  | Bit 12: IE\_NETWORK\_QOS |
|  |  | Bit 13: IE\_NETWORK\_DATA\_RATE |
|  |  | Bit 14: IE\_NET\_REGULT\_DOMAIN |
|  |  | Bit 15: IE\_NET\_FREQUENCY\_BANDS |
|  |  | Bit 16: IE\_NET\_IP\_CFG\_METHODS |
|  |  | Bit 17: IE\_NET\_CAPABILITIES |
|  |  | Bit 18: IE\_NET\_SUPPORTED\_LCP |
|  |  | Bit 19: IE\_NET\_MOB\_MGMT\_PROT |
|  |  | Bit 20: IE\_NET\_EMSERV\_PROXY |
|  |  | Bit 21: IE\_NET\_IMS\_PROXY\_CSCF |
|  |  | Bit 22: IE\_NET\_MOBILE\_NETWORK |
|  |  | Bit 23: IE\_POA\_LINK\_ADDR |
|  |  | Bit 24: IE\_POA\_LOCATION |
|  |  | Bit 25: IE\_POA\_CHANNEL\_RANGE |
|  |  | Bit 26: IE\_POA\_SYSTEM\_INFO |
|  |  | Bit 27: IE\_POA\_SUBNET\_INFO |
|  |  | Bit 28: IE\_POA\_IP\_ADDR |
|  |  | Bit 29–63: (Reserved) |
|  |  |  |

**Table F.20—Data type for MIS capabilities *(continued)***

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| MIS\_TRANS\_LST | BITMAP(16) | A list of supported transports.  Bitmap Values:  Bit 0: UDP  Bit 1: TCP  Bit 2–15: (Reserved) |
| NET\_TYPE\_ADDR | SEQUENCE(  NETWORK\_TYPE, LINK\_ADDR  ) | Represent a link address of a specific network type. |

### Data type for MIS registration

Table F.21—Data type for MIS registration

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| REG\_REQUEST\_CODE | ENUMERATED | The registration code: 0—Registration  1—Re-Registration |

### Data types for handover operation

Table F.22—Data type for handover operation

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| ASGN\_RES\_SET | SEQUENCE (  QOS\_LIST,  TSP\_CONTAINER  ) | Set of resource parameters reserved and assigned by the target network to the MN for performing handover to a network PoA. The transparent con­tainer is from target to source, which includes the required configuration of the reserved resources at the target network. |
| HO\_CAUSE | UNSIGNED\_INT(1) | Represents the reason for performing a handover.  Same enumeration list as link down reason code. See Table F.7. |
| HO\_RESULT | ENUMERATED | Handover result.  0: Success   1. Failure 2. Rejected |
| HO\_STATUS | ENUMERATED | Represents the permission for handover.  0: HandoverPerMISed 1: HandoverDeclined |
| PREDEF\_CFG\_ID | INTEGER(1) | Pre-defined configuration identifier. 0..255 |

**Table F.22—Data type for handover operation *(continued)***

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| RQ\_RESULT | SEQUENCE(  LINK\_POA\_LIST,  QOS\_LIST, CHOICE(NULL,BOOLEAN, IP\_CFG\_MTHDS),  CHOICE(NULL, BOOLEAN, DHCP\_SERV),  CHOICE(NULL, BOOLEAN, FN\_AGNT),  CHOICE(NULL, BOOLEAN, ACC\_RTR)  ) | Represents the result of network resource query. The LINK\_POA\_LIST is a list of potential PoAs for a given link with the same IP configuration information.  The list is sorted from most preferred first to least preferred last.  The QOS\_LIST contains the available resources for this list of PoAs.  BOOLEAN value is set to TRUE when the requested value is the same as compared to the value transmitted in the request.  BOOLEAN value can only be set when the request contained the corresponding IPConfigura­tionMethods, DHCPServerAddress, FAAddress, or AccessRouterAddress information. |
| DHCP\_SERV | IP\_ADDR | IP address of candidate DHCP Server. It is only included when dynamic address configuration is supported. |
| FN\_AGNT | IP\_ADDR | IP address of candidate Foreign Agent. It is only included when Mobile IPv4 is supported. |
| ACC\_RTR | IP\_ADDR | IP address of candidate Access Router. It is only included when IPv6 Stateless configuration is sup­ported. |
| REQ\_RES\_SET | SEQUENCE(QOS\_LIST, TSP\_CONTAINER,  HO\_CAUSE  ) | Set of resource parameters required for perform­ing admission control and resource reservation for the MN at the target network. The transparent container is from source to target, which includes the required MN configuration for adMISing the new connection at the target network and reserv­ing resources. |
| TGT\_NET\_INFO | CCHOICE ( SEQUENCE(NETWORK\_ID, CHOICE(NULL, NET\_AUX\_ID)), LINK\_ADDR  ) | Represents the handover commit information. LINK\_ADDR is the target PoA link address. |
| TSP\_CARRIER | OCTET\_STRING | Transparent carrier containing link specific infor­mation whose content and format are to be speci­fied by the link specific SDO. |
| TSP\_CONTAINER | CHOICE (  NULL,  PREDEF\_CFG\_ID, TSP\_CARRIER  ) | Transparent container. If the value is null, this parameters is not available. |

### Data type for MIS\_NET\_SAP primitives

Table F.23—Data type for MIS\_NET\_SAP primitives

|  |  |  |
| --- | --- | --- |
| **Data type name** | **Derived from** | **Definition** |
| TRANSPORT\_TYPE | ENUMERATED | The transport type supported: 0: L2  1: L3 or higher layer protocols |

**F.3.16 Data type for security**

Table 29 (F.24) Data type for security

|  |  |  |
| --- | --- | --- |
| **Data type** | **Derived from** | **Definition** |
| ID\_TYPE | EUMERATED | The type of security association:  0: TLS-generated;  1: EAP-generated |
| ID\_VALUE | OCTET\_STRING | Represents a security association identifier |
| MIS\_SEC\_CAP | SEQUENCE( TLS\_CAP, EAP\_CAP) | Represents the MIS security capabilities. |
| TLS\_CAP | BOOLEAN | TLS-generated SA capability. TRUE: (D)TLS Supported FALSE: (D)TLS not supported |
| EAP\_CAP | CHOICE( NULL,SEQUENCE (KEY\_DIST\_LIST, INT\_ALG\_LIST, CIPH\_ALG\_LIST, PRF\_LIST)) | EAP-generated SA capability. When NULL is chosen, EAP-generated SA is not supported. When SEQUENCE is chosen, EAP-generated SA is supported. |
| KEY\_DIST\_LIST | BITMAP(B) | Represents a list of key distribution methods. Bitmap values  Bit 0: Push key distribution  Bit 1: Optimized proactive pull key distribution  Bit 2: Reactive pull key distribution  Bit 3–7 (Reserved) |
| INT\_ALG\_LIST | BITMAP(B) | Represents a list of integrity algorithm. Bitmap values  Bit 0: INTG\_HMAC\_SHA\_96  Bit 1: INTG\_HMAC\_CMAC\_AES Bit 2–7 (Reserved) |
| CIPH\_ALG\_LIST | BITMAP(B) | Represents a list of encryption algorithm. Bitmap values  Bit 0: ENCR\_AES\_CBC  Bit 1: AUTH\_ENC\_AES\_CCM Bit 2: ENCR\_NULL  Bit 3-7 (reserved) |
| PRF\_LIST | BITMAP(B) | Represents a list of key derivation functions. Bitmap values  Bit 0: PRF\_AES\_CMAC  Bit 1: PRF\_HMAC\_SHA1  Bit 2: PRF\_HMAC\_SHA256  Bit 3-7 (reserved) |
| NONCE\_VALUE | UNSIGNED\_INT(2) | Represents a random value |
| AUTH\_INFO\_VALUE | OCTET\_STRING | Represents the authentication information used to authenticate. |
| AUTH\_VALUE | OCTET\_STRING | Represents a message authentication/integrity code. |
| KEY | OCTET\_STRING | Represents a cryptographic key. |
| KEY\_MAPPING | LIST(SEQUENCE(LINK\_TUPLE\_ ID, KEY, LIFETIME)) | Represents a map of a link layer identifier of a  PoA to a key and a lifetime. |
| LINK\_AUTHENTICA TOR\_LIST | SEQUENCE(LINK\_TYPE, LIST(LINK\_ADDR)) | Represents a list of link layer addresses of au- thenticators for a given link type. |
| LL\_FRAMES | OCTET\_STRING | Represents the information needed to carry out a key installation. |
| LIFETIME | UNSIGNED\_INT(2) | Represents the period of time that a key is valid and can be used. |
| SECURITY | CHOICE (TLS\_RECORD, MIS\_SPS\_RECORD) | Represents information which is carried in the security TLV. |
| TLS\_RECORD | OCTET\_STRING | Represents a TLS record. |
| MIS\_SPS\_RECORD | SEQUENCE (ENCR\_BLOCK, CHOICE (INTG\_BLOCK, NULL)) | Represents data protected by an MIS security association. |
| ENCR\_BLOCK | OCTET\_STRING | Represents encrypted data. |
| INTG\_BLOCK | OCTET\_STRING | Represents integrity protected data. |

# (normative) Information element identifiers

Table G.1 lists the information element identifier values for different individual IEs and IE containers.

Table G.1—Information element identifier values

|  |  |
| --- | --- |
| **Name of information element or container** | **IE Identifier** |
| IE\_NETWORK\_TYPE | 0x10000000 |
| IE\_OPERATOR\_ID | 0x10000001 |
| IE\_SERVICE\_PROVIDER\_ID | 0x10000002 |
| IE\_COUNTRY\_CODE | 0x10000003 |
| IE\_NETWORK\_ID | 0x10000100 |
| IE\_NETWORK\_AUX\_ID | 0x10000101 |
| IE\_ROAMING\_PARTNERS | 0x10000102 |
| IE\_COST | 0x10000103 |
| IE\_NETWORK\_QOS | 0x10000105 |
| IE\_NETWORK\_DATA\_RATE | 0x10000106 |
| IE\_NET\_REGULAT\_DOMAIN | 0x10000107 |
| IE\_NET\_FREQUENCY\_BANDS | 0x10000108 |
| IE\_NET\_IP\_CFG\_METHODS | 0x10000109 |
| IE\_NET\_CAPABILITIES | 0x1000010A |
| IE\_NET\_SUPPORTED\_LCP | 0x1000010B |
| IE\_NET\_MOB\_MGMT\_PROT | 0x1000010C |
| IE\_NET\_EMSERV\_PROXY | 0x1000010D |
| IE\_NET\_IMS\_PROXY\_CSCF | 0x1000010E |
| IE\_NET\_MOBILE\_NETWORK | 0x1000010F |
| IE\_POA\_LINK\_ADDR | 0x10000200 |
| IE\_POA\_LOCATION | 0x10000201 |
| IE\_POA\_CHANNEL\_RANGE | 0x10000202 |
| IE\_POA\_SYSTEM\_INFO | 0x10000203 |
| IE\_POA\_SUBNET\_INFO | 0x10000204 |
| IE\_POA\_IP\_ADDR | 0x10000205 |
| IE\_CONTAINER\_LIST\_OF\_NETWORKS | 0x10000300 |
| IE\_CONTAINER\_NETWORK | 0x10000301 |
| IE\_CONTAINER\_POA | 0x10000302 |
| IE\_AUTHENTICATOR\_LINK\_ADDR | 0x10000206 |
| IE\_AUTHENTICATOR\_IP\_ADDR | 0x10000207 |
| IE\_PoS\_IP\_ADDR | 0x10000208 |

# (normative) MIIS basic schema

The following text defines the RDF vocabularies for MIIS.

<?xml version="1.0"?>

<!DOCTYPE rdf:RDF [

<!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#">

<!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">

<!ENTITY misbasic "URL\_TO\_BE\_ASSIGNED">

<!ENTITY owl "http://www.w3.org/2002/07/owl#">

<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">

]>

<rdf:RDF xmlns:rdf="&rdf;" xmlns:rdfs="&rdfs;"

xmlns:misbasic="&misbasic;" xml:base="&misbasic;"

xmlns:owl="&owl;" xmlns:xsd="&xsd;">

<owl:Ontology rdf:about="">

<rdfs:label>

Basic Schema for IEEE 802.21 Information Service

</rdfs:label>

<owl:versionInfo>1.0</owl:versionInfo>

</owl:Ontology>

<owl:DatatypeProperty rdf:ID="ie\_identifier">

<rdfs:subPropertyOf rdf:resource="&rdfs;label"/>

<rdfs:range rdf:resource="&xsd;hexBinary"/>

<rdfs:comment>

A type identifier values for Information Elements.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="bit\_number">

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

<rdfs:comment>

This property represents a bit number that has

the value as true.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_container\_list\_of\_networks">

<misbasic:ie\_identifier>0x10000300</misbasic:ie\_identifier>

<rdfs:range rdf:resource="#LIST\_OF\_NETWORKS"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="LIST\_OF\_NETWORKS">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_container\_network"/>

<owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:minCardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:ObjectProperty rdf:ID="ie\_container\_network">

<misbasic:ie\_identifier>0x10000301</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#LIST\_OF\_NETWORKS"/>

<rdfs:range rdf:resource="#NETWORK"/>

<rdfs:comment>

This class contains General Information depicting and Access

Network Specific Information.

</rdfs:comment>

</owl:ObjectProperty>

<owl:Class rdf:ID="NETWORK">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_network\_type"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_operator\_id"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:ObjectProperty rdf:ID="ie\_network\_type">

<misbasic:ie\_identifier>0x10000000</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="#NETWORK\_TYPE"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="NETWORK\_TYPE">

</owl:Class>

<owl:DatatypeProperty rdf:ID="link\_type">

<rdfs:domain rdf:resource="#NETWORK\_TYPE"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

<rdfs:comment>

Link type of a network. The following values are assigned:

1: Wireless - GSM

2: Wireless - GPRS

3: Wireless - EDGE

15: Ethernet

18: Wireless - Other

19: Wireless - IEEE 802.11

22: Wireless - CDMA2000

23: Wireless - UMTS

24: Wireless - cdma-2000-HRPD

27: Wireless - IEEE 802.16

28: Wireless - IEEE 802.20

29: Wireless - IEEE 802.22

</rdfs:comment>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="subtype">

<rdfs:subPropertyOf rdf:resource="#bit\_number"/>

<rdfs:domain rdf:resource="#NETWORK\_TYPE"/>

<rdfs:comment>

The range of #bit\_number is 0-63.

</rdfs:comment>

</owl:ObjectProperty>

<owl:DatatypeProperty rdf:ID="type\_ext">

<rdfs:domain rdf:resource="#NETWORK\_TYPE"/>

<rdfs:range rdf:resource="&xsd;string"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_operator\_id">

<misbasic:ie\_identifier>0x10000001</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="#OPERATOR\_ID"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="OPERATOR\_ID">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#op\_name"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#op\_namespace"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:DatatypeProperty rdf:ID="op\_name">

<rdfs:domain rdf:resource="#OPERATOR\_ID"/>

<rdfs:range rdf:resource="&xsd;string"/>

<rdfs:comment>

The value is a non NULL terminated

string whose length shall not exceed 253 octets.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="op\_namespace">

<rdfs:domain rdf:resource="#OPERATOR\_ID"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

<rdfs:comment>

A value of Operator Type:

0: GSM/UMTS

1: CDMA

2: REALM

3: ITU-T/TSB

4: General

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="ie\_service\_provider\_id">

<misbasic:ie\_identifier>0x10000002</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="&xsd;string"/>

<rdfs:comment>

A non-NULL terminated string whose length shall not exceed 253 octets.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="ie\_country\_code">

<misbasic:ie\_identifier>0x10000003</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="&xsd;string"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="ie\_network\_id">

<misbasic:ie\_identifier>0x10000100</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="&xsd;string"/>

<rdfs:comment>

A non-NULL terminated string whose length shall not exceed 253 octets.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="ie\_network\_aux\_id">

<misbasic:ie\_identifier>0x10000101</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="&xsd;string"/>

<rdfs:comment>

It is HESSID if network type is IEEE 802.11.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_roaming\_partners">

<misbasic:ie\_identifier>0x10000102</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="#OPERATOR\_ID"/>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="ie\_cost">

<misbasic:ie\_identifier>0x10000103</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="#COST"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="COST">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#cost\_unit"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#cost\_value"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#cost\_curr"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:DatatypeProperty rdf:ID="cost\_unit">

<rdfs:domain rdf:resource="#COST"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

<rdfs:comment>

The unit of the cost:

0: second

1: minute

2: hours

3: day

4: week

5: month

6: year

7: free

8: flat rate

9-255: Reserved

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="cost\_value">

<rdfs:domain rdf:resource="#COST"/>

<rdfs:range rdf:resource="&xsd;double"/>

<rdfs:comment>

The cost value in Currency/Unit

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="cost\_curr">

<rdfs:domain rdf:resource="#COST"/>

<rdfs:range rdf:resource="&xsd;string"/>

<rdfs:comment>

A three-letter currency code(e.g., "USD") specified by

ISO 4217.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_network\_qos">

<misbasic:ie\_identifier>0x10000105</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="#QOS\_LIST"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="QOS\_LIST">

</owl:Class>

<owl:DatatypeProperty rdf:ID="num\_cos\_types">

<rdfs:domain rdf:resource="#QOS\_LIST"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

</owl:DatatypeProperty>

<owl:Class rdf:ID="COS">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#cos\_id"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#cos\_value"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:DatatypeProperty rdf:ID="cos\_id">

<rdfs:domain rdf:resource="#COS"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

<rdfs:comment>

A type to represent a class of service identifier.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="cos\_value">

<rdfs:domain rdf:resource="#COS"/>

<rdfs:range rdf:resource="&xsd;unsignedShort"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="min\_pk\_tx\_delay">

<rdfs:domain rdf:resource="#QOS\_LIST"/>

<rdfs:range rdf:resource="#COS"/>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="avg\_pk\_tx\_delay">

<rdfs:domain rdf:resource="#QOS\_LIST"/>

<rdfs:range rdf:resource="#COS"/>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="max\_pk\_tx\_delay">

<rdfs:domain rdf:resource="#QOS\_LIST"/>

<rdfs:range rdf:resource="#COS"/>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="pk\_delay\_jitter">

<rdfs:domain rdf:resource="#QOS\_LIST"/>

<rdfs:range rdf:resource="#COS"/>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="pk\_loss\_rate">

<rdfs:domain rdf:resource="#QOS\_LIST"/>

<rdfs:range rdf:resource="#COS"/>

</owl:ObjectProperty>

<owl:DatatypeProperty rdf:ID="ie\_network\_data\_rate">

<misbasic:ie\_identifier>0x10000106</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="&xsd;unsignedInt"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_net\_regulat\_domain">

<misbasic:ie\_identifier>0x10000107</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="#REGU\_DOMAIN"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="REGU\_DOMAIN">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#regu\_domain\_country\_code"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#regu\_class"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:DatatypeProperty rdf:ID="regu\_domain\_country\_code">

<rdfs:domain rdf:resource="#REGU\_DOMAIN"/>

<rdfs:range rdf:resource="&xsd;String"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="regu\_class">

<rdfs:domain rdf:resource="#REGU\_DOMAIN"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="ie\_net\_frequency\_bands">

<misbasic:ie\_identifier>0x10000108</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="&xsd;unsignedInt"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_net\_ip\_cfg\_methods">

<misbasic:ie\_identifier>0x10000109</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="#IP\_CONFIG"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="IP\_CONFIG">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ip\_cfg\_mthds"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:DatatypeProperty rdf:ID="ip\_cfg\_mthds">

<rdfs:subPropertyOf rdf:resource="#bit\_number"/>

<rdfs:domain rdf:resource="#IP\_CONFIG"/>

<rdfs:comment>

The range of #bit\_number is 0-31.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="dhcp\_serv">

<rdfs:domain rdf:resource="#IP\_CONFIG"/>

<rdfs:range rdf:resource="#TRANSPORT\_ADDR"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="fn\_agnt">

<rdfs:domain rdf:resource="#IP\_CONFIG"/>

<rdfs:range rdf:resource="#TRANSPORT\_ADDR"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="acc\_rtr">

<rdfs:domain rdf:resource="#IP\_CONFIG"/>

<rdfs:range rdf:resource="#TRANSPORT\_ADDR"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_net\_capabilities">

<misbasic:ie\_identifier>0x1000010A</misbasic:ie\_identifier>

<rdfs:subPropertyOf rdf:resource="#bit\_number"/>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:comment>

The range of #bit\_number is 0-31.

</rdfs:comment>

</owl:ObjectProperty>

<owl:DatatypeProperty rdf:ID="ie\_net\_supported\_lcp">

<misbasic:ie\_identifier>0x1000010B</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_net\_mob\_mgmt\_prot">

<misbasic:ie\_identifier>0x1000010C</misbasic:ie\_identifier>

<rdfs:subPropertyOf rdf:resource="#bit\_number"/>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:comment>

The range of #bit\_number is 0-15.

</rdfs:comment>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="ie\_net\_emserv\_proxy">

<misbasic:ie\_identifier>0x1000010D</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="#PROXY\_ADDR"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="PROXY\_ADDR">

</owl:Class>

<owl:DatatypeProperty rdf:ID="proxy\_addr\_ip">

<rdfs:domain rdf:resource="#PROXY\_ADDR"/>

<rdfs:range rdf:resource="#TRANSPORT\_ADDR"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="proxy\_addr\_fqdn">

<rdfs:domain rdf:resource="#PROXY\_ADDR"/>

<rdfs:range rdf:resource="&xsd;String"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_net\_ims\_proxy\_cscf">

<misbasic:ie\_identifier>0x1000010E</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="#PROXY\_ADDR"/>

</owl:ObjectProperty>

<owl:DatatypeProperty rdf:ID="ie\_net\_mobile\_network">

<misbasic:ie\_identifier>0x1000010F</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="&xsd;boolean"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_container\_poa">

<misbasic:ie\_identifier>0x10000302</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#NETWORK"/>

<rdfs:range rdf:resource="#POA"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="POA">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_poa\_link\_addr"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_poa\_location"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_poa\_channel\_range"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_poa\_system\_info"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_poa\_subnet\_info"/>

<owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:minCardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_authenticator\_link\_addr"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_authenticator\_ip\_addr"/>

<owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:minCardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ie\_pos\_ip\_addr"/>

<owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:minCardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:comment>

This class contains all the information depicting a PoA.

</rdfs:comment>

</owl:Class>

<owl:ObjectProperty rdf:ID="ie\_poa\_link\_addr">

<misbasic:ie\_identifier>0x10000200</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#POA"/>

<rdfs:range rdf:resource="#LINK\_ADDR"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="LINK\_ADDR">

</owl:Class>

<owl:DatatypeProperty rdf:ID="mac\_addr">

<rdfs:domain rdf:resource="#LINK\_ADDR"/>

<rdfs:range rdf:resource="#TRANSPORT\_ADDR"/>

</owl:DatatypeProperty>

<owl:Class rdf:ID="LINK\_ADDR\_3GPP\_3G">

<rdfs:subClassOf rdf:resource="#LINK\_ADDR"/>

</owl:Class>

<owl:DatatypeProperty rdf:ID="link\_addr\_3gpp\_3g\_cell\_id\_plmn\_id">

<rdfs:domain rdf:resource="#LINK\_ADDR\_3GPP\_3G"/>

<rdfs:range rdf:resource="&xsd;string"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="link\_addr\_3gpp\_3g\_cell\_id\_cell\_id">

<rdfs:domain rdf:resource="#LINK\_ADDR\_3GPP\_3G"/>

<rdfs:range rdf:resource="&xsd;unsignedInt"/>

</owl:DatatypeProperty>

<owl:Class rdf:ID="LINK\_ADDR\_3GPP\_2G">

<rdfs:subClassOf rdf:resource="#LINK\_ADDR"/>

</owl:Class>

<owl:DatatypeProperty rdf:ID="link\_addr\_3gpp\_2g\_cell\_id\_plmn\_id">

<rdfs:domain rdf:resource="#LINK\_ADDR\_3GPP\_2G"/>

<rdfs:range rdf:resource="&xsd;string"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="link\_addr\_3gpp\_2g\_cell\_id\_lac">

<rdfs:domain rdf:resource="#LINK\_ADDR\_3GPP\_2G"/>

<rdfs:range rdf:resource="&xsd;string"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="link\_addr\_3gpp\_2g\_cell\_id\_ci">

<rdfs:domain rdf:resource="#LINK\_ADDR\_3GPP\_2G"/>

<rdfs:range rdf:resource="&xsd;string"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="\_3gpp\_addr">

<rdfs:domain rdf:resource="#LINK\_ADDR"/>

<rdfs:range rdf:resource="&xsd;string"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="\_3gpp2\_addr">

<rdfs:domain rdf:resource="#LINK\_ADDR"/>

<rdfs:range rdf:resource="&xsd;string"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="link\_addr\_other\_l2\_addr">

<rdfs:domain rdf:resource="#LINK\_ADDR"/>

<rdfs:range rdf:resource="&xsd;string"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_poa\_location">

<misbasic:ie\_identifier>0x10000201</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#POA"/>

<rdfs:range rdf:resource="#LOCATION"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="LOCATION">

</owl:Class>

<owl:Class rdf:ID="BIN\_GEO\_LOC">

<rdfs:subClassOf rdf:resource="#LOCATION"/>

<rdfs:comment>

This class has properties that represent geographic coordinate.

The format is based on the Location Configuration Information (LCI)

defined in RFC 3825.

</rdfs:comment>

</owl:Class>

<owl:DatatypeProperty rdf:ID="la\_res">

<rdfs:domain rdf:resource="#BIN\_GEO\_LOC"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="latitude">

<rdfs:domain rdf:resource="#BIN\_GEO\_LOC"/>

<rdfs:range rdf:resource="&xsd;double"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="lo\_res">

<rdfs:domain rdf:resource="#BIN\_GEO\_LOC"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="longitude">

<rdfs:domain rdf:resource="#BIN\_GEO\_LOC"/>

<rdfs:range rdf:resource="&xsd;double"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="at">

<rdfs:domain rdf:resource="#BIN\_GEO\_LOC"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

<rdfs:comment>

Following codes are defined:

1: Meters: in 2s-complement fixed-point 22-bit integer part with

8-bit fraction. If AT = 1, an AltRes value 0.0 would indicate

unknown altitude. The most precise Altitude would have an AltRes

value of 30. Many values of AltRes would obscure any variation

due to vertical datum differences.

2: Floors: in 2s-complement fixed-point 22-bit integer part with

8-bit fraction. AT = 2 for Floors enables representing altitude in

a form more relevant in buildings which have different

floor-to-floor dimensions.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="alt\_res">

<rdfs:domain rdf:resource="#BIN\_GEO\_LOC"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

<rdfs:comment>

Altitude resolution: 6 bits indicating the number of valid bits

in the altitude. Values above 30 (decimal) are undefined and

reserved.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="altitude">

<rdfs:domain rdf:resource="#BIN\_GEO\_LOC"/>

<rdfs:range rdf:resource="&xsd;double"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="datum">

<rdfs:domain rdf:resource="#BIN\_GEO\_LOC"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

<rdfs:comment>

Following codes are defined:

1: WGS

2: NAD 83 (with associated vertical datum for North American

vertical datum for 1998)

3: NAD 83 (with associated vertical datum for Mean Lower Low Water

(MLLW))

</rdfs:comment>

</owl:DatatypeProperty>

<owl:Class rdf:ID="XML\_GEO\_LOC">

<rdfs:subClassOf rdf:resource="#LOCATION"/>

</owl:Class>

<owl:DatatypeProperty rdf:ID="xml\_geo\_loc">

<rdfs:domain rdf:resource="#XML\_GEO\_LOC"/>

<rdfs:range rdf:resource="&xsd;string"/>

<rdfs:comment>

Geo address elements as described in RFC4119.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:Class rdf:ID="BIN\_CIVIC\_LOC">

<rdfs:subClassOf rdf:resource="#LOCATION"/>

<rdfs:comment>

This class has properties that represent civic address.

The format is defined in IETF RFC 4676.

</rdfs:comment>

</owl:Class>

<owl:DatatypeProperty rdf:ID="civic\_cntry\_code">

<rdfs:domain rdf:resource="#BIN\_CIVIC\_LOC"/>

<rdfs:range rdf:resource="&xsd;string"/>

<rdfs:comment>

Two-letter ISO 3166 country code in capital ASCII letters.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="civic\_addr">

<rdfs:domain rdf:resource="#BIN\_CIVIC\_LOC"/>

<rdfs:range rdf:resource="#CIVIC\_ADDR"/>

<rdfs:comment>

This property contains the civic address elements.

The format of the civic address elements is described

in Section 3.4 of IETF RFC 4676 with a TLV pair

(whereby the Type and Length fields are one octet long).

</rdfs:comment>

</owl:ObjectProperty>

<owl:Class rdf:ID="CIVIC\_ADDR">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#catype"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#cavalue"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:DatatypeProperty rdf:ID="catype">

<rdfs:domain rdf:resource="#CIVIC\_ADDR"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

<rdfs:comment>

A one-octet descriptor of the data civic address value.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="cavalue">

<rdfs:domain rdf:resource="#CIVIC\_ADDR"/>

<rdfs:range rdf:resource="&xsd;string"/>

<rdfs:comment>

The civic address value.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:Class rdf:ID="XML\_CIVIC\_LOC">

<rdfs:subClassOf rdf:resource="#LOCATION"/>

</owl:Class>

<owl:DatatypeProperty rdf:ID="xml\_civic\_loc">

<rdfs:domain rdf:resource="#XML\_CIVIC\_LOC"/>

<rdfs:range rdf:resource="&xsd;string"/>

<rdfs:comment>

Geo address elements as described in RFC4119.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:Class rdf:ID="LOCATION\_CELL\_ID">

<rdfs:subClassOf rdf:resource="#LOCATION"/>

</owl:Class>

<owl:DatatypeProperty rdf:ID="location\_cell\_id">

<rdfs:domain rdf:resource="#LOCATION\_CELL\_ID"/>

<rdfs:range rdf:resource="&xsd;unsignedInt"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_poa\_channel\_range">

<misbasic:ie\_identifier>0x10000202</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#POA"/>

<rdfs:range rdf:resource="#CH\_RANGE"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="CH\_RANGE">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#low\_ch\_range"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#high\_ch\_range"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:DatatypeProperty rdf:ID="low\_ch\_range">

<rdfs:domain rdf:resource="#CH\_RANGE"/>

<rdfs:range rdf:resource="&xsd;unsignedInt"/>

<rdfs:comment>

Lowest channel frequency in MHz

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="high\_ch\_range">

<rdfs:domain rdf:resource="#CH\_RANGE"/>

<rdfs:range rdf:resource="&xsd;unsignedInt"/>

<rdfs:comment>

Highest channel frequency in MHz

</rdfs:comment>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_poa\_system\_info">

<misbasic:ie\_identifier>0x10000203</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#POA"/>

<rdfs:range rdf:resource="#SYSTEM\_INFO"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="SYSTEM\_INFO">

</owl:Class>

<owl:ObjectProperty rdf:ID="system\_info\_network\_type">

<rdfs:domain rdf:resource="#SYSTEM\_INFO"/>

<rdfs:range rdf:resource="#NETWORK\_TYPE"/>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="system\_info\_link\_addr">

<rdfs:domain rdf:resource="#SYSTEM\_INFO"/>

<rdfs:range rdf:resource="#LINK\_ADDR"/>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="system\_info\_parameters">

<rdfs:domain rdf:resource="#SYSTEM\_INFO"/>

<rdfs:range rdf:resource="#PARAMETERS"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="PARAMETERS">

</owl:Class>

<owl:Class rdf:ID="DCD\_UCD">

<rdfs:subClassOf rdf:resource="#PARAMETERS"/>

</owl:Class>

<owl:DatatypeProperty rdf:ID="dcd\_ucd\_base\_id">

<rdfs:domain rdf:resource="#DCD\_UCD"/>

<rdfs:range rdf:resource="&xsd;unsignedShort"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="bandwidth">

<rdfs:domain rdf:resource="#DCD\_UCD"/>

<rdfs:range rdf:resource="&xsd;unsignedShort"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="du\_ctr\_freq">

<rdfs:domain rdf:resource="#DCD\_UCD"/>

<rdfs:range rdf:resource="&xsd;unsignedLong"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="eirp">

<rdfs:domain rdf:resource="#DCD\_UCD"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="ttg">

<rdfs:domain rdf:resource="#DCD\_UCD"/>

<rdfs:range rdf:resource="&xsd;unsignedShort"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="rtg">

<rdfs:domain rdf:resource="#DCD\_UCD"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="down\_bp">

<rdfs:subPropertyOf rdf:resource="#bit\_number"/>

<rdfs:domain rdf:resource="#DCD\_UCD"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="up\_bp">

<rdfs:subPropertyOf rdf:resource="#bit\_number"/>

<rdfs:domain rdf:resource="#DCD\_UCD"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="init\_code">

<rdfs:domain rdf:resource="#DCD\_UCD"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="ho\_code">

<rdfs:domain rdf:resource="#DCD\_UCD"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

</owl:DatatypeProperty>

<owl:Class rdf:ID="SIB">

<rdfs:subClassOf rdf:resource="#PARAMETERS"/>

</owl:Class>

<owl:DatatypeProperty rdf:ID="sib\_cell\_id">

<rdfs:domain rdf:resource="#SIB"/>

<rdfs:range rdf:resource="&xsd;unsignedInt"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="fq\_code\_num">

<rdfs:domain rdf:resource="#SIB"/>

<rdfs:range rdf:resource="&xsd;unsignedShort"/>

</owl:DatatypeProperty>

<owl:Class rdf:ID="SYS\_PARAMS">

<rdfs:subClassOf rdf:resource="#PARAMETERS"/>

</owl:Class>

<owl:DatatypeProperty rdf:ID="sys\_params\_base\_id">

<rdfs:domain rdf:resource="#SYS\_PARAMS"/>

<rdfs:range rdf:resource="&xsd;unsignedShort"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="pilot\_pn">

<rdfs:domain rdf:resource="#SYS\_PARAMS"/>

<rdfs:range rdf:resource="&xsd;unsignedShort"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="freq\_id">

<rdfs:domain rdf:resource="#SYS\_PARAMS"/>

<rdfs:range rdf:resource="&xsd;unsignedShort"/>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="band\_class">

<rdfs:domain rdf:resource="#SYS\_PARAMS"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_poa\_subnet\_info">

<misbasic:ie\_identifier>0x10000204</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#POA"/>

<rdfs:range rdf:resource="#IP\_SUBNET\_INFO"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="IP\_SUBNET\_INFO">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#ip\_prefix\_len"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#subnet\_address"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:DatatypeProperty rdf:ID="ip\_prefix\_len">

<rdfs:domain rdf:resource="#IP\_SUBNET\_INFO"/>

<rdfs:range rdf:resource="&xsd;unsignedByte"/>

<rdfs:comment>

The bit length of the prefix of the subnet to which subnet\_address

property belongs. The prefix\_length is less than or equal to 32

for IPv4 subnet and less than or equal to 128 for IPv6 subnet.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="subnet\_address">

<rdfs:domain rdf:resource="#IP\_SUBNET\_INFO"/>

<rdfs:range rdf:resource="#TRANSPORT\_ADDR"/>

<rdfs:comment>

An IP address of the PoA encoded as Address base type defined in

RFC3588. The first 2-octet contains AddressType, which may be

either 1 (IPv4) or 2 (IPv6). If AddressType==1, the subnet\_address

property contains a 4-octet IPv4 address. If AddressType==2, the

subnet\_address property contains a 16-octet IPv6 address.

</rdfs:comment>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="ie\_poa\_ip\_addr">

<misbasic:ie\_identifier>0x10000205</misbasic:ie\_identifier>

<rdfs:domain rdf:resource="#POA"/>

<rdfs:range rdf:resource="#TRANSPORT\_ADDR"/>

</owl:ObjectProperty>

<owl:Class rdf:ID="TRANSPORT\_ADDR">

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#addr\_family"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

<rdfs:subClassOf>

<owl:Restriction>

<owl:onProperty rdf:resource="#addr\_value"/>

<owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1

</owl:cardinality>

</owl:Restriction>

</rdfs:subClassOf>

</owl:Class>

<owl:DatatypeProperty rdf:ID="addr\_family">

<rdfs:domain rdf:resource="#TRANSPORT\_ADDR"/>

<rdfs:range rdf:resource="&xsd;unsignedShort"/>

<rdfs:comment>

An Address Family defined in

http://www.iana.org/assignments/address-family-numbers.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="addr\_value">

<rdfs:domain rdf:resource="#TRANSPORT\_ADDR"/>

<rdfs:range rdf:resource="&xsd;hexBinary"/>

<rdfs:comment>

An address value specific to address\_type.

</rdfs:comment>

</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="ie\_authenticator\_link\_addr">

<misbasic:ie\_type\_identifier>0x10000206</misbasic:ie\_type\_identifier>

<rdfs:domain rdf:resource="#POA"/>

<rdfs:range rdf:resource="#TRANSPORT\_ADDR"/>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="ie\_authenticator\_ip\_addr">

<misbasic:ie\_type\_identifier>0x10000207</misbasic:ie\_type\_identifier>

<rdfs:domain rdf:resource="#POA"/>

<rdfs:range rdf:resource="#TRANSPORT\_ADDR"/>

</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="ie\_pos\_ip\_addr">

<misbasic:ie\_type\_identifier>0x10000208</misbasic:ie\_type\_identifier>

<rdfs:domain rdf:resource="#POA"/>

<rdfs:range rdf:resource="#TRANSPORT\_ADDR"/>

</owl:ObjectProperty>

</rdf:RDF>

# (informative) Making user extensions to MIIS schema

This annex describes how to create an extended schema. How to create “IP address of Mobile IP home agent” is used as an example.

It is possible to support an Extended Schema without defining additional IEs or TLVs (including vendor specific ones) other than those that are defined in this standard. An extended schema can be defined as an XML document. An example Extended Schema definition is shown as follows.

<?xml version="1.0"?>

<!DOCTYPE rdf:RDF [

<!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#">

<!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">

<!ENTITY misbasic "URL\_TO\_BE\_ASSIGNED">

<!ENTITY misextended "http://www.informationservice.org/2006/08/extended-schema#">

<!ENTITY owl "http://www.w3.org/2002/07/owl#">

<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">

]>

<rdf:RDF xmlns:rdf="&rdf;" xmlns:rdfs="&rdfs;" xmlns:misbasic="&misbasic;"

xmlns:misextended="&misextended;" xml:base="&misextended;" xmlns:owl="&owl;" xmlns:xsd="&xsd;">

<owl:Ontology rdf:about="">

<rdfs:label>Extended Schema</rdfs:label>

</owl:Ontology>

<owl:ObjectProperty rdf:ID="ha\_address">

<rdfs:domain rdf:resource="&misbasic;NETWORK"/>

<rdfs:range rdf:resource="&misbasic;TRANSPORT\_ADDR"/>

</owl:ObjectProperty>

</rdf:RDF>

# (normative) IEEE 802.21 MIB

## Parameters requiring MIB definition

In this standard, only parameters that govern the operation of IEEE 802.21 are defined.

### MIS capability parameters

The following is a list of parameters that are used in the MIS capability discovery:

1. **Link Address List (Read-only)**: A list of network type and link address pairs that are controlled by this MISF. Note that not all interfaces of an MIS-capable node can be under control of MISF.
2. **MIS Event List (Read-only)**: A list of supported events by this MISF.
3. **MIS Command List (Read-only)**: A list of supported commands by this MISF.
4. **MIS IS Query List (Read-only)**: A list of supported MIS IS query types by this MISF.
5. **MIS Transport List (Read-only)**: A list of supported MIS transport protocols by this MISF. This is the transport that transmits the MIS protocol messages.
6. **List of MBB Handover support (Read-only)**: A list of make-before-break support information for each pair of serving and target network types.

### MIS protocol parameters

The following is a list of parameters that are used in the MIS protocol:

1. Local MISF ID (Read-Write)
2. List of Peer MISF IDs (Read-Write)
3. Transport Type (Read-Write)

* L2 or L3
* Defined for each peer MISF

1. Version (Read-only)
2. Maximum Transaction Lifetime (Read-Write)

* Unit: seconds

1. Maximum Retransmission Interval (Read-Write)

* Unit: seconds

1. Maximum Retransmission Counter (Read-Write)

* Unit: none

1. Maximum Average Transmission Rate (Read-Write)

* Maximum value of average transmission rate
* Unit: octets per second

1. Maximum Burst Size (Read-Write) — The maximum number of octets transmitted in a burst

* Unit: octets

1. aFragmentationThreshold (Read-Write)

* Unit: octets

1. ReassemblyTimer (Read-Write)

* Unit: seconds

## IEEE 802.21 MIB definition

IEEE802dot21-MIB DEFINITIONS ::= BEGIN

IMPORTS

MODULE-IDENTITY, OBJECT-TYPE, Unsigned32 FROM SNMPv2-SMI

MODULE-COMPLIANCE, OBJECT-GROUP FROM SNMPv2-CONF

TEXTUAL-CONVENTION, TruthValue FROM SNMPv2-TC;

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \* MODULE IDENTITY

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ieee802dot21 MODULE-IDENTITY

LAST-UPDATED "200806041455Z"

ORGANIZATION "IEEE 802.21"

CONTACT-INFO

"WG E-mail: stds-802-21@ieee.org

Chair: Subir Das

Telcordia Technologies

E-mail: [subir@research.telcordia.com](mailto:subir@research.telcordia.com)

Editor: David Cypher

E-mail: [david.cypher@nist.gov](mailto:david.cypher@nist.gov)"

DESCRIPTION

"The MIB module for IEEE 802.21 entities.

iso(1).std(0).iso8802(8802).ieee802dot21(21)"

REVISION "201105161205Z"

DESCRIPTION

"The latest version of this MIB module."

::= { iso std(0) iso8802(8802) ieee802dot21(21) }

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \* Textual Conventions

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Dot21MisfID ::= TEXTUAL-CONVENTION

DISPLAY-HINT "253a"

STATUS current

DESCRIPTION

"The MISF ID of an MIS node."

REFERENCE "IEEE Std 802.21, 2008 Edition, F.3.11"

SYNTAX OCTET STRING (SIZE(0..253))

Dot21LinkType ::= TEXTUAL-CONVENTION

DISPLAY-HINT "d"

STATUS current

DESCRIPTION

"This attribute represents the type of a link."

REFERENCE "IEEE Std 802.21, 2008 Edition, F.3.4"

SYNTAX Unsigned32 (0..255)

Dot21NetworkSubtype ::= TEXTUAL-CONVENTION

DISPLAY-HINT "8x"

STATUS current

DESCRIPTION

"This attribute represents the network subtype of a link."

REFERENCE "IEEE Std 802.21, 2008 Edition, F.3.8"

SYNTAX OCTET STRING (SIZE(0..8))

Dot21NetworkTypeExtension ::= TEXTUAL-CONVENTION

DISPLAY-HINT "253a"

STATUS current

DESCRIPTION

"This attribute represents a network type extension."

REFERENCE "IEEE Std 802.21, 2008 Edition, F.3.8"

SYNTAX OCTET STRING (SIZE(0..253))

Dot21EventList ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"This attribute represents a list of supported events."

REFERENCE "IEEE Std 802.21, 2008 Edition, F.3.12"

SYNTAX BITS

{ misLinkDetected(0),

misLinkUp(1),

misLinkDown(2),

misLinkParametersReport(3),

misLinkGoingDown(4),

misLinkHandoverImminent(5),

misLinkHandoverComplete(6),

misLinkPDUTransmitStatus(7) }

Dot21CommandList ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"This attribute represents a list of supported commands."

REFERENCE "IEEE Std 802.21, 2008 Edition, F.3.12"

SYNTAX BITS

{ misGetLinkParameters(0),

misLinkConfigureThresholds(1),

misLinkActions(2),

misNetworkHandoverCommands(3),

misMobileHandoverCommands(4) }

Dot21ISQueryTypeList ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

" This attribute will be a set of supported MIS IS query types."

REFERENCE "IEEE Std 802.21, 2008 Edition, F.3.12"

SYNTAX BITS

{ binary(0),

rdfData(1),

rdfSchemaUrl(2),

rdfSchema(3),

typeIeNetworkType(4),

typeIeOperatorIdentifier(5),

typeIeServiceProviderIdentifier(6),

typeIeCountryCode(7),

typeIeNetworkIdentifier(8),

typeIeNetworkAuxiliaryIdentifier(9),

typeIeRoamingPartners(10),

typeIeCost(11),

typeIeNetworkQos(12),

typeIeNetworkDataRate(13),

typeIeNetworkRegulatoryDomain(14),

typeIeNetworkFrequencyBands(15),

typeIeNetworkIpConfigurationMethods(16),

typeIeNetworkCapabilities(17),

typeIeNetworkSupportedLcp(18),

typeIeNetworkMobilityManagementProtocol(19),

typeIeNetworkEmergencyServiceProxy(20),

typeIeNetworkImsProxyCscf(21),

typeIeNetworkMobileNetwork(22),

typeIePoaLinkAddress(23),

typeIePoaLocation(24),

typeIePoaChannelRange(25),

typeIePoaSystemInformation(26),

typeIePoaSubnetInformation(27),

typeIePoaIpAddress(28),

typeIeAuthenticatorLinkAddress(29),

typeIeAutheticatorIpAddress(30),

typeIePosIpAddress(31) }

Dot21TransportList ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

" This attribute will be a set of supported MIS transports."

REFERENCE "IEEE Std 802.21, 2008 Edition, F.3.12"

SYNTAX BITS { udp(0), tcp(1) }

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \* Major sections

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- MIS Function Management (MISMT) Attributes

-- DEFINED AS "The MISMT object class provides the necessary support

-- at the MISF to manage the processes in the station such that

-- the MISF can work cooperatively as a part of an IEEE 802.21

-- network."

dot21mismt OBJECT IDENTIFIER ::= { ieee802dot21 1 }

-- dot21mismt GROUPS

-- dot21LocalMisfTable ::= { dot21mismt 1 }

-- dot21PeerMisfTable ::= { dot21mismt 2 }

-- dot21MbbHandoverSupportTable ::= { dot21mismt 3 }

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \* MIB attribute OBJECT-TYPE definitions follow

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

--

-- Local MISF Table

--

dot21LocalMisfTable OBJECT-TYPE

SYNTAX SEQUENCE OF Dot21LocalMisfEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The table of local MISFs. The MIS MIB allows to have more than one local MISFs per SNMP engine."

::={ dot21mismt 1 }

dot21LocalMisfEntry OBJECT-TYPE

SYNTAX Dot21LocalMisfEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The value contains information associated with a particular local MISF. In most cases, there will be only one local MISF on a node."

INDEX { dot21LocalMisfIndex }

::={ dot21LocalMisfTable 1}

Dot21LocalMisfEntry ::=

SEQUENCE{

dot21LocalMisfIndex Unsigned32,

dot21LocalMisfID Dot21MisfID,

dot21LocalEventList Dot21EventList,

dot21LocalCommandList Dot21CommandList,

dot21LocalISQueryTypeList Dot21ISQueryTypeList,

dot21LocalTransportList Dot21TransportList,

dot21LocalVersion Unsigned32,

dot21LocalMaxTransactionLifetime Unsigned32,

dot21LocalMaxRetransmissionIntvl Unsigned32,

dot21LocalMaxRetransmissionCntr Unsigned32,

dot21LocalMaxAvgTransmissionRate Unsigned32,

dot21LocalMaxBurstSize Unsigned32,

dot21LocalFragmentationThreshold Unsigned32,

dot21LocalReassemblyTimeout Unsigned32

}

dot21LocalMisfIndex OBJECT-TYPE

SYNTAX Unsigned32(0..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Index of local MISF table."

::= { dot21LocalMisfEntry 1 }

dot21LocalMisfID OBJECT-TYPE

SYNTAX Dot21MisfID

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The MISF ID of this node."

REFERENCE "IEEE Std 802.21, 2008 Edition, F.3.11"

::={ dot21LocalMisfEntry 2 }

dot21LocalEventList OBJECT-TYPE

SYNTAX Dot21EventList

MAX-ACCESS read-only

STATUS current

DESCRIPTION

" This attribute will be a set of all the MIS events supported by this MIS node."

DEFVAL { {} }

::={ dot21LocalMisfEntry 3 }

dot21LocalCommandList OBJECT-TYPE

SYNTAX Dot21CommandList

MAX-ACCESS read-only

STATUS current

DESCRIPTION

" This attribute will be a set of all the MIS commands supported by this MIS node."

DEFVAL { {} }

::={ dot21LocalMisfEntry 4 }

dot21LocalISQueryTypeList OBJECT-TYPE

SYNTAX Dot21ISQueryTypeList

MAX-ACCESS read-only

STATUS current

DESCRIPTION

" This attribute will be a set of MIS IS query types supported by this MIS node."

DEFVAL { {} }

::={ dot21LocalMisfEntry 5 }

dot21LocalTransportList OBJECT-TYPE

SYNTAX Dot21TransportList

MAX-ACCESS read-only

STATUS current

DESCRIPTION

" This attribute will be a set of MIS transports supported by this MIS node."

DEFVAL { {} }

::={ dot21LocalMisfEntry 6 }

dot21LocalVersion OBJECT-TYPE

SYNTAX Unsigned32 (1..15)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The MIS protocol version supported by this MISF."

DEFVAL { 1 }

::={ dot21LocalMisfEntry 7 }

dot21LocalMaxTransactionLifetime OBJECT-TYPE

SYNTAX Unsigned32 (1..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum time in seconds for an MIS protocol transaction."

DEFVAL { 30 }

::={ dot21LocalMisfEntry 8 }

dot21LocalMaxRetransmissionIntvl OBJECT-TYPE

SYNTAX Unsigned32 (1..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum time in seconds for retransmitting an MIS message."

DEFVAL { 10 }

::={ dot21LocalMisfEntry 9 }

dot21LocalMaxRetransmissionCntr OBJECT-TYPE

SYNTAX Unsigned32 (1..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum number of retransmission retries for MIS messages."

DEFVAL { 2 }

::={ dot21LocalMisfEntry 10 }

dot21LocalMaxAvgTransmissionRate OBJECT-TYPE

SYNTAX Unsigned32 (0..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum number of MIS messages can be transmitted per second on this node. If the value is 0, no

limitation is set."

DEFVAL { 0 }

::={ dot21LocalMisfEntry 11 }

dot21LocalMaxBurstSize OBJECT-TYPE

SYNTAX Unsigned32 (0..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

" The maximum number of octets transmitted in a burst. If the value is 0, no limitation is set."

DEFVAL { 0 }

::={ dot21LocalMisfEntry 12 }

dot21LocalFragmentationThreshold OBJECT-TYPE

SYNTAX Unsigned32 (8..65535)

MAX-ACCESS read-write

STATUS current

DESCRIPTION "The value for aFragmentationThreshold on this node."

DEFVAL { 1500 }

::={ dot21LocalMisfEntry 13 }

dot21LocalReassemblyTimeout OBJECT-TYPE

SYNTAX Unsigned32 (1..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION "The timeout value for ReassemblyTimer."

DEFVAL { 5 }

::={ dot21LocalMisfEntry 14 }

--

-- The Peer MISF Table

--

dot21PeerMisfTable OBJECT-TYPE

SYNTAX SEQUENCE OF Dot21PeerMisfEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The table of MISF known by this MISF."

::={ dot21mismt 2 }

dot21PeerMisfEntry OBJECT-TYPE

SYNTAX Dot21PeerMisfEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Details of a specific MISF peer."

INDEX {dot21PeerMisfIndex}

::= { dot21PeerMisfTable 1 }

Dot21PeerMisfEntry ::=

SEQUENCE {

dot21PeerMisfIndex Unsigned32,

dot21PeerMisfID Dot21MisfID,

dot21PeerLocalMisfID Dot21MisfID,

dot21PeerEventList Dot21EventList,

dot21PeerCommandList Dot21CommandList,

dot21PeerISQueryTypeList Dot21ISQueryTypeList,

dot21PeerTransportList Dot21TransportList,

dot21PeerTransportType INTEGER,

dot21PeerVersion Unsigned32,

dot21PeerMaxTransactionLifetime Unsigned32,

dot21PeerMaxRetransmissionIntvl Unsigned32,

dot21PeerMaxRetransmissionCntr Unsigned32,

dot21PeerMaxAvgTransmissionRate Unsigned32,

dot21PeerMaxBurstSize Unsigned32,

dot21PeerFragmentationThreshold Unsigned32,

dot21PeerReassemblyTimeout Unsigned32

}

dot21PeerMisfIndex OBJECT-TYPE

SYNTAX Unsigned32(0..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Index of peer MISF table."

::= { dot21PeerMisfEntry 1 }

dot21PeerMisfID OBJECT-TYPE

SYNTAX Dot21MisfID

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The MISF ID of a peer MIS node."

::={ dot21PeerMisfEntry 2 }

dot21PeerLocalMisfID OBJECT-TYPE

SYNTAX Dot21MisfID

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The MISF ID of the local MIS node for this peer MIS node."

::={ dot21PeerMisfEntry 3 }

dot21PeerEventList OBJECT-TYPE

SYNTAX Dot21EventList

MAX-ACCESS read-only

STATUS current

DESCRIPTION

" This attribute will be a set of all the MIS events supported by peer MIS node."

DEFVAL { {} }

::={ dot21PeerMisfEntry 4 }

dot21PeerCommandList OBJECT-TYPE

SYNTAX Dot21CommandList

MAX-ACCESS read-only

STATUS current

DESCRIPTION

" This attribute will be a set of all the MIS commands supported by peer MIS node."

DEFVAL { {} }

::={ dot21PeerMisfEntry 5 }

dot21PeerISQueryTypeList OBJECT-TYPE

SYNTAX Dot21ISQueryTypeList

MAX-ACCESS read-only

STATUS current

DESCRIPTION

" This attribute will be a set of MIS IS query types supported by peer MIS node."

DEFVAL { {} }

::={ dot21PeerMisfEntry 6 }

dot21PeerTransportList OBJECT-TYPE

SYNTAX Dot21TransportList

MAX-ACCESS read-only

STATUS current

DESCRIPTION

" This attribute will be a set of MIS transports supported by peer MIS node."

DEFVAL { {} }

::={ dot21PeerMisfEntry 7 }

dot21PeerTransportType OBJECT-TYPE

SYNTAX INTEGER { layerTwo(2), layerThree(3) }

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This value should be set for the MIS protocol layer used for transmitting MIS messages."

DEFVAL { layerTwo }

::= {dot21PeerMisfEntry 8 }

dot21PeerVersion OBJECT-TYPE

SYNTAX Unsigned32 (1..15)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The MIS protocol version supported by peer MISF. The default version is 1."

DEFVAL { 1 }

::={ dot21PeerMisfEntry 9 }

dot21PeerMaxTransactionLifetime OBJECT-TYPE

SYNTAX Unsigned32 (1..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum time in seconds for an MIS protocol transaction used for a particular peer MISF."

DEFVAL { 30 }

::={ dot21PeerMisfEntry 10 }

dot21PeerMaxRetransmissionIntvl OBJECT-TYPE

SYNTAX Unsigned32 (1..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum time in seconds for retransmitting an MIS message used for a particular peer MISF."

DEFVAL { 10 }

::={ dot21PeerMisfEntry 11 }

dot21PeerMaxRetransmissionCntr OBJECT-TYPE

SYNTAX Unsigned32 (1..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum number of retransmission retries for MIS messages used for a particular peer MISF."

DEFVAL { 2 }

::={ dot21PeerMisfEntry 12 }

dot21PeerMaxAvgTransmissionRate OBJECT-TYPE

SYNTAX Unsigned32 (0..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum number of MIS messages can be transmitted per second on this node for a particular peer MISF.

If the value is 0, no limitation is set."

DEFVAL { 0 }

::={ dot21PeerMisfEntry 13 }

dot21PeerMaxBurstSize OBJECT-TYPE

SYNTAX Unsigned32 (0..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum number of octets transmitted in a burst. If the value is 0, no limitation is set."

DEFVAL { 0 }

::={ dot21PeerMisfEntry 14 }

dot21PeerFragmentationThreshold OBJECT-TYPE

SYNTAX Unsigned32 (8..65535)

MAX-ACCESS read-write

STATUS current

DESCRIPTION "The value for aFragmentationThreshold used for this peer MIS node."

DEFVAL { 1500 }

::={ dot21PeerMisfEntry 15 }

dot21PeerReassemblyTimeout OBJECT-TYPE

SYNTAX Unsigned32 (1..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION "The timeout value for ReassemblyTimer used for this peer MIS node."

DEFVAL { 5 }

::={ dot21PeerMisfEntry 16 }

--

-- The Make-Before-Break Handover Support Table

--

dot21MbbHandoverSupportTable OBJECT-TYPE

SYNTAX SEQUENCE OF Dot21MbbHandoverSupportEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The table of make-before-break handover support entries."

::={ dot21mismt 4 }

dot21MbbHandoverSupportEntry OBJECT-TYPE

SYNTAX Dot21MbbHandoverSupportEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The value contains information associated with a particular MBB support."

INDEX { dot21MbbHandoverSupportIndex }

::={ dot21MbbHandoverSupportTable 1 }

Dot21MbbHandoverSupportEntry ::=

SEQUENCE{

dot21MbbHandoverSupportIndex Unsigned32,

dot21FromLinkType Dot21LinkType,

dot21FromNetworkSubtype Dot21NetworkSubtype,

dot21FromNetworkTypeExtension Dot21NetworkTypeExtension,

dot21ToLinkType Dot21LinkType,

dot21ToNetworkSubtype Dot21NetworkSubtype,

dot21ToNetworkTypeExtension Dot21NetworkTypeExtension,

dot21IsMbbSupported TruthValue

}

dot21MbbHandoverSupportIndex OBJECT-TYPE

SYNTAX Unsigned32(0..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Index of make-before-break handover support table."

::= { dot21MbbHandoverSupportEntry 1 }

dot21FromLinkType OBJECT-TYPE

SYNTAX Dot21LinkType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This attribute represents the link type of serving link."

DEFVAL { 0 }

::={ dot21MbbHandoverSupportEntry 2 }

dot21FromNetworkSubtype OBJECT-TYPE

SYNTAX Dot21NetworkSubtype

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This attribute represents the network subtype of serving link."

DEFVAL { ''H }

::={ dot21MbbHandoverSupportEntry 3 }

dot21FromNetworkTypeExtension OBJECT-TYPE

SYNTAX Dot21NetworkTypeExtension

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This attribute represents the network type extension of serving link."

DEFVAL { ''H }

::={ dot21MbbHandoverSupportEntry 4 }

dot21ToLinkType OBJECT-TYPE

SYNTAX Dot21LinkType

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This attribute represents the link type of target link."

DEFVAL { 0 }

::={ dot21MbbHandoverSupportEntry 5 }

dot21ToNetworkSubtype OBJECT-TYPE

SYNTAX Dot21NetworkSubtype

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This attribute represents the network subtype of target link."

DEFVAL { ''H }

::={ dot21MbbHandoverSupportEntry 6 }

dot21ToNetworkTypeExtension OBJECT-TYPE

SYNTAX Dot21NetworkTypeExtension

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This attribute represents the network type extension of target link."

DEFVAL { ''H }

::={ dot21MbbHandoverSupportEntry 7 }

dot21IsMbbSupported OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This attribute indicates whether make-before-break handover is supported. A value of true indicates that make-before-break handover is supported. A value of FALSE indicates that make-before-break handover is not supported."

::={ dot21MbbHandoverSupportEntry 8 }

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \* Conformance Information

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

dot21Conformance OBJECT IDENTIFIER ::= { ieee802dot21 2 }

dot21Groups OBJECT IDENTIFIER ::= { dot21Conformance 1 }

dot21Compliances OBJECT IDENTIFIER ::= { dot21Conformance 2 }

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- \* Compliance Statements

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

dot21Compliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION

"The compliance statement for SNMPv2 entities that implement the IEEE 802.21 MIB."

MODULE -- this module

MANDATORY-GROUPS {

dot21MismtBase1

}

::= { dot21Compliances 1 }

dot21MismtBase1 OBJECT-GROUP

OBJECTS {

dot21LocalMisfID,

dot21LocalEventList,

dot21LocalCommandList,

dot21LocalISQueryTypeList,

dot21LocalTransportList,

dot21LocalVersion,

dot21LocalMaxTransactionLifetime,

dot21LocalMaxRetransmissionIntvl,

dot21LocalMaxRetransmissionCntr,

dot21LocalMaxAvgTransmissionRate,

dot21LocalMaxBurstSize,

dot21LocalFragmentationThreshold,

dot21LocalReassemblyTimeout,

dot21PeerMisfID,

dot21PeerLocalMisfID,

dot21PeerEventList,

dot21PeerCommandList,

dot21PeerISQueryTypeList,

dot21PeerTransportList,

dot21PeerTransportType,

dot21PeerVersion,

dot21PeerMaxTransactionLifetime,

dot21PeerMaxRetransmissionIntvl,

dot21PeerMaxRetransmissionCntr,

dot21PeerMaxAvgTransmissionRate,

dot21PeerMaxBurstSize,

dot21PeerFragmentationThreshold,

dot21PeerReassemblyTimeout,

dot21FromLinkType,

dot21FromNetworkSubtype,

dot21FromNetworkTypeExtension,

dot21ToLinkType,

dot21ToNetworkSubtype,

dot21ToNetworkTypeExtension,

dot21IsMbbSupported

}

STATUS current

DESCRIPTION

"This object class provides the necessary support at the MIS node to manage the processes in the MIS node, so that the MIS node may work cooperatively as a part of an IEEE 802.21 network."

::= { dot21Groups 1 }

END

# Fragmentation (informative)

## Example MIS message fragmentation

An example of an original MIS message and fragmented MIS messages is shown in Figure K. 1.

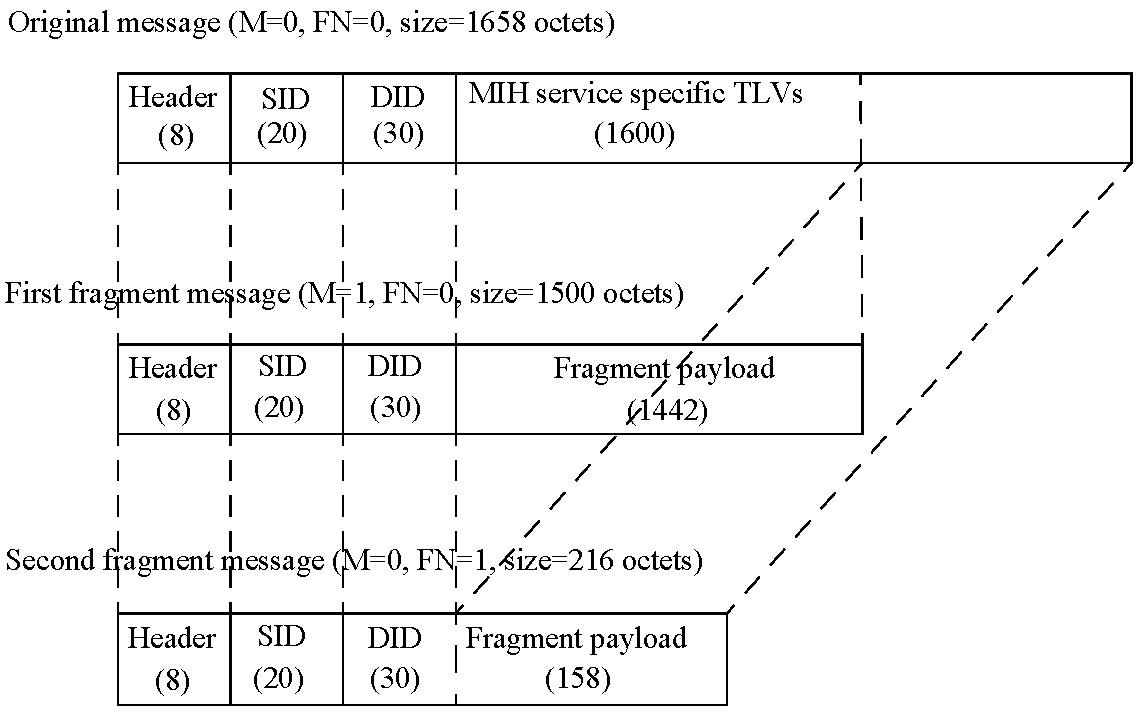


Figure K.1—MIS Fragmentation example for MTU of 1500 octets

## Calculation of securityOverhead when there is an MIS SA

To calculate securityOverhead when there is an MIS SA, the following parameters are used:

— x is 0 when Source MISF Identifier TLV and Destination MISF Identifier TLV are contained in the protected MIS message, otherwise, x is 1.

— y is 1 for TLS-generated MIS SA. Otherwise, y is 0.

— LSAID denotes the octet length of the SAID TLV carried in the protected MIS message. LSAID

depends on the implementation.

— LSID denotes the octet length of the Source MISF Identifier TLV optionally carried in the protected

MIS message. LSID depends on the implementation.

— LDID denotes the octet length of the Destination MISF Identifier TLV optionally carried in the pro- tected MIS message. LDID depends on the implementation.

— OSECTLV denotes the overhead of the Security TLV carried in the protected MIS message.

— OTYPE(y) denotes the overhead of the MIS data type contained in the Security TLV.

— OTLS denotes the overhead of the TLS record. OTLS = 5, i.e., 1-octet TLSCiphertext.type plus 2-octet

TLSCiphertext.version plus 2-octet TLSCiphertext.length [RFC5246].

— OENC denotes the overhead of encryption. OENC depends on the ciphersuite.

— OINTG denotes the overhead of integrity protection. OINTG depends on the ciphersuite. securityOverhead is calculated as follows:

securityOverhead = LSAID –x\*(LSID + LDID)+ OSECTLV + OTYPE(y) + y\*OTLS + OENC + OINTG

Note that securityOverhead can be a negative value when x = 1.

Since the maximum size of Security TLV is no more than the maximum size of Variable Payload of MIS

message, which is 216–1 octets, the maximum values of OSECTLV and OTYPE(y) are shown below.

— OSECTLV = 3 (i.e., 1-octet TLV Type plus 2-octet TLV Length).

— OTYPE(0) = 6, i.e., 1-octet CHOICE Selector in CHOICE(TLS\_RECORD, MIS\_SPS\_RECORD) plus 2-octet Length field of ENCR\_BLOCK data plus 1-octet CHOICE Selector in MIS\_SPS\_RECORD plus 2- octet Length field of INTG\_BLOCK data.

— OTYPE(1) = 3, i.e., 1-octet CHOICE Selector in CHOICE(TLS\_RECORD, MIS\_SPS\_RECORD)

plus 2-octet Length field of TLS\_RECORD data.

Table K.1 shows OENC and OINTG values for the MIS ciphersuites for EAP-generated MIS SA.

**Table K.1—Protection Overhead for EAP-generated SAs**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ciphersuite code** | **Encryption** | **Integrity** **Ptotection** | **OENC** | **OINTG** |
| 00000010 | AES\_CBC | HMAC-SHA1-96 | 32(IV+padding) | 12 (MIC) |
| 00000100 | NULL | HMAC-SHA1-96 | 0 | 12 (MIC) |
| 00000101 | NULL | AES\_CMAC | 0 | 12 (MIC) |
| 00000110 | AES\_CCM | | 10(SN)+ 12(MIC) | 0 |

For example, consider a case where Ciphersuite Code 00000010 (AES-CBC + HMAC-SHA1-96) is used for EAP-generated MIS SA (y=0) without containing Source MISF Identifier TLV and Destination MISF Identifier TLV in the protected MIS message (x=0), and the length of SAID TLV, the length of Source MISF Identifier TLV, the length of Destination MISF Identifier TLV are 30 octets, 20 octets and 30 octets, respectively. Then securityOverhead is computed as:

securityOverhead = LSAID – (LSID + LDID) + OSECTLV + OTYPE(0) + OENC + OINTG

= 30– (20+30)+3+6+44 = 33 (octets).

Figure K.2 shows the protected fragments for the original message shown in Figure K.1, when operating in the same condition as described in the above example with securityOverhead=33 (octets). The integer number within the brackets of each field in Figure K.2 indicates the length of the field in octets. In Figure K.2, the fragment size before applying MIS protection is set to 1424 (=16\*89) octets to have the fragment size of 1499 octets after applying MIS protection, which gives the largest number of 16-octet blocks (89) under the condition that the resulting protected fragment does not exeeds 1500 octets.

First protected fragment message (M=1, FN=0, size =1499 octets)

Header (S=1) (8)

SAID TLV (30)

Security TLV (1461)

Encrypted fragment = 16\*19 = 1424 octets

IV = 16 octets

MIC = 12 octets

TLV overhead = 3 octets

MIS data type overhead = 6 octets

Second protected fragment message (M=0, FN=1, size = 251 octets)

Header (S=1) (8)

SAID TLV (30)

Security TLV (213)

Encrypted fragment = 1600-1424 = 176 octets

IV = 16 octets

MIC = 12 octets

TLV overhead = 3 octets

MIS data type overhead = 6 octets

**Figure K.2—Example of protected MIS fragment message**

# (normative) MIS protocol message code assignments

Table L.1 provides the action identifier (AID) assignment for MIS messages.

Table L.1—AID assignment

|  |  |
| --- | --- |
| **MIS messages** | **AID** |
| *MIS messages for Service Management* | |
| MIS\_Capability\_Discover | 1 |
| MIS\_Register | 2 |
| MIS\_DeRegister | 3 |
| MIS\_Event\_Subscribe | 4 |
| MIS\_Event\_Unsubscribe | 5 |
| MIS\_Auth | 6 |
| MIS\_Termination\_Auth | 7 |
| MIS\_Push\_Key | 8 |
| MIS\_LL\_Auth | 9 |
| *MIS messages for Event Service* | |
| MIS\_Link\_Detected | 1 |
| MIS\_Link\_Up | 2 |
| MIS\_Link\_Down | 3 |
| MIS\_Link\_Parameters\_Report | 5 |
| MIS\_Link\_Going\_Down | 6 |
| MIS\_Link\_TTandover\_Imminent | 7 |
| MIS\_Link\_TTandover\_Complete | 8 |
| *MIS messages for Command Service* | |
| MIS\_Link\_Get\_Parameters | 1 |
| MIS\_Link\_Configure\_Thresholds | 2 |
| MIS\_Link\_Actions | 3 |
| MIS\_Net\_TTO\_Candidate\_Query | 4 |
| MIS\_MN\_TTO\_Candidate\_Query | 5 |
| MIS\_N2N\_TTO\_Query\_Resources | 6 |
| MIS\_MN\_TTO\_Commit | 7 |
| MIS\_Net\_TTO\_Commit | 8 |
| MIS\_N2N\_TTO\_Commit | 9 |
| MIS\_MN\_TTO\_Complete | 10 |
| MIS\_N2N\_TTO\_Complete | 11 |

**Table L.1—AID assignment *(continued)***

|  |  |
| --- | --- |
| **MIS messages** | **AID** |
| *MIS messages for Information Service* | |
| MIS\_Get\_Information | 1 |
| MIS\_Push\_Information | 2 |

Table L.2 provides the TLV type value assignment for MIS messages. The type value can be extracted from the binary encoding method of the corresponding data type. TLV type value 101–255 is reserved for experi­mental TLVs.

Table L.2—Type values for TLV encoding

|  |  |  |
| --- | --- | --- |
| **TLV type name** | **TLV type value** | **Data type** |
| Source MISF ID | 1 | MISF\_ID |
| Destination MISF ID | 2 | MISF\_ID |
| Status | 3 | STATUS |
| Link type | 4 | LINK\_TYPE |
| MIS event list | 5 | MIS\_EVT\_LIST |
| MIS command list | 6 | MIS\_CMD\_LIST |
| MIIS query type list | 7 | MIS\_IQ\_TYPE\_LST |
| Transport option list | 8 | MIS\_TRANS\_LST |
| Link address list | 9 | LIST(NET\_TYPE\_ADDR) |
| MBB handover support | 10 | LIST(MBB\_HO\_SUPP) |
| Register request code | 11 | REG\_REQUEST\_CODE |
| Valid time interval | 12 | UNSIGNED\_INT(4) |
| Link identifier | 13 | LINK\_TUPLE\_ID |
| New link identifier | 14 | LINK\_TUPLE\_ID |
| Old access router | 15 | LINK\_ADDR |
| New access router | 16 | LINK\_ADDR |
| IP renewal flag | 17 | IP\_RENEWAL\_FLAG |
| Mobility management support | 18 | IP\_MOB\_MGMT |
| IP address configuration methods | 19 | IP\_CFG\_MTHDS |
| Link down reason code | 20 | LINK\_DN\_REASON |
| Time interval | 21 | UNSIGNED\_INT(2) |
| Link going down reason | 22 | LINK\_GD\_REASON |
| Link parameter report list | 23 | LIST(LINK\_PARAM\_RPT) |

**Table L.2—Type values for TLV encoding *(continued)***

|  |  |  |
| --- | --- | --- |
| **TLV type name** | **TLV type value** | **Data type** |
| Device states request | 24 | DEV\_STATES\_REQ |
| Link identifier list | 25 | LIST(LINK\_ID) |
| Device states response list | 26 | LIST(DEV\_STATES\_RSP) |
| Get status request set | 27 | LINK\_STATUS\_REQ |
| Get status response list | 28 | LIST( SEQUENCE(LINK\_ID,LINK\_STATUS\_RSP) ) |
| Configure request list | 29 | LIST(LINK\_CFG\_PARAM) |
| Configure response list | 30 | LIST(LINK\_CFG\_STATUS) |
| List of link PoA list | 31 | LIST(LINK\_POA\_LIST) |
| Preferred link list | 32 | LIST(RQ\_RESULT) |
| Handover resource query list | 33 | QOS\_LIST |
| Handover status | 34 | HO\_STATUS |
| Access router address | 35 | IP\_ADDR |
| DHCP server address | 36 | IP\_ADDR |
| FA address | 37 | IP\_ADDR |
| Link actions list | 38 | LIST(LINK\_ACTION\_REQ) |
| Link actions result list | 39 | LIST(LINK\_ACTION\_RSP) |
| Handover result | 40 | HO\_RESULT |
| Resource status | 41 | LINK\_RES\_STATUS |
| Resource retention status | 42 | BOOLEAN |
| Info query binary data list | 43 | LIST(IQ\_BIN\_DATA) |
| Info query RDF data list | 44 | LIST(IQ\_RDF\_DATA) |
| Info query RDF schema URL | 45 | BOOLEAN |
| Info query RDF schema list | 46 | LIST(IQ\_RDF\_SCHM) |
| Max response size | 47 | UNSIGNED\_INT(2) |
| Info response binary data list | 48 | LIST(IR\_BIN\_DATA) |
| Info response RDF data list | 49 | LIST(IR\_RDF\_DATA) |
| Info response RDF schema URL list | 50 | LIST(IR\_SCHM\_URL) |
| Info response RDF schema list | 51 | LIST(IR\_RDF\_SCHM) |
| MN MISF ID | 52 | MISF\_ID |
| Query resource report flag | 53 | BOOLEAN |
| Event configuration info list | 54 | LIST(EVT\_CFG\_INFO) |

**Table L.2—Type values for TLV encoding *(continued)***

|  |  |  |
| --- | --- | --- |
| **TLV type name** | **TLV type value** | **Data type** |
| Target network info | 55 | TGT\_NET\_INFO |
| List of target network info | 56 | LIST(TGT\_NET\_INFO) |
| Assigned resource set | 57 | ASGN\_RES\_SET |
| Link detected info list | 58 | LIST(LINK\_DET\_INFO) |
| MN link ID | 59 | LINK\_ID |
| PoA | 60 | LINK\_ADDR |
| Unauthenticated information request | 61 | BOOLEAN |
| Network type | 62 | NETWORK\_TYPE |
| Requested resource set | 63 | REQ\_RES\_SET |
| Security | 64 | SECURITY |
| SAID | 65 | SEQUENCE(ID\_TYPE, ID\_VALUE) |
| Security capability | 66 | MIS\_SEC\_CAP |
| KeyLifeTime | 67 | LIFETIME |
| AUTH | 68 | AUTH\_VALUE |
| NONCE | 69 | NONCE\_VALUE |
| Authentication | 70 | AUTH\_INFO\_VALUE |
| Link identifier | 71 | LINK\_TUPLE \_ID |
| Link layer information | 72 | LL\_FRAME |
| Link Tuple Identifier List | 73 | LIST(LINK\_TUPLE\_ID) |
| Authenticator list | 74 | LIST(LINK\_AUTHENTICATOR\_LIST) |
| Ciphersuite | 75 | SEQUENCE(KEY\_DIST\_LIST, INT\_ALG\_LIST, CIPH\_ALG\_LIST, PRF\_LIST) |
| (Reserved) | 80–99 | (Reserved) |
| Vendor specific TLV | 100 | (Vendor specific) |
| (Reserved for experimental TLVs) | 10 1–255 | (Used for experimental purposes) |

# (normative) Protocol implementation conformance statement (PICS) proforma

## Introduction

To evaluate conformance of a particular implementation, it is necessary to have a statement of which capabilities and options have been implemented for a given IEEE 802 standard. Such a statement is called an Implementation Conformance Statement (ICS).

## Scope

This annex provides the ICS proforma for IEEE Std 802.21 in compliance with the relevant requirements, and in accordance with the relevant guidance, given in ITU-T Recommendation X.290 (1995), OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications—General concept, and.ITU-T Recommendation X.296 (1995) OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications—Implementation conformance statements.

## Conformance

If it is claimed to conform to IEEE Std 802.21, the actual PICS Proforma to be filled in by a supplier shall be technically equivalent to the text of the PICS Proforma in this annex, and shall preserve the number/naming and ordering of the PICS Proforma items.

A PICS that conforms to this IEEE Std 802.21 shall be a conforming PICS proforma completed in accordance with the instructions for completion given in M.4.

## Instructions

### Purpose and structure

The supplier of a protocol implementation that is claimed to conform to IEEE Std 802.21, shall complete the following protocol implementation conformance statement (PICS) proforma.

The PICS proforma expresses in compact form the static conformance requirements of this standard. It serves as a reference to the static conformance review. ITU-T Recommendation X.296, 6.7, provides examples of uses and users of proformas.

A completed PICS proforma is the PICS for the implementation in question. The PICS is a statement of which capabilities and options of the protocol have been implemented.

14Copyright release for PICS proformas: Users of this standard may freely reproduce the PICS proforma in this annex so that it can be used for its intended purpose and may further publish the completed PICS.

This PICS proforma has the following structure. Within this, the instructions subclause, are the purpose and scope; symbols, abbreviations, and terms; and explicit instructions for completing the implementation conformance statement. After the instructions are the subclauses for identifying the implementation, the protocol, and corrigenda (if any). The final subclause contains the questionnaire in tabular form. Within this final subclause a separate table is used to cover these categorizes: global statement of conformance, roles, major capabilities, protocol data units (PDUs), PDU parameters, and timers.

The structure of the individual tables varies. Except for the tables for the identification of the protocol and the identification of any corrigenda, at a minimum all tables have columns for item number, item description, status, and support. For most the status columns includes both a status and a predicate. Some tables have a column for a mnemonic, which makes for easier cross-referencing within the PICS proforma. Most tables contain a reference column. A few tables (e.g., PDU parameters and timers) contain a column for entering supported value(s).

### Symbols, abbreviations, and terms

M mandatory

O optional

O.<n> optional, but support of at least one of the group of options labeled by the same numeral <n> is required

pred: conditional symbol, including predicate identification (mnemonic)

N/A not applicable

GBLx mnemonic for global statement of conformance, where x is an integer

MCx mnemonic for major capabilities, where x is an integer

PDUx mnemonic for protocol data unit, where x is an integer

RLx mnemonic for role, where x is an integer

### Explicit instructions

The blank spaces in the identification of the implementation part is to be completed with the information necessary to identify fully both the implementation and supplier, as well as the name of a person to contact if there are any queries concerning the contents of the PICS.

For the identification of the protocol, indicate in the support column “Yes,” if this is the protocol being supported.

For the identification of corrigenda to the protocol, indicate if any corrigenda have been applied by entering the corrigenda information in the space provided. If none are applicable, then leave it blank.

The main part of the PICS proforma is a fixed questionnaire in tabular form, divided into subclauses, each containing a number of individual items. Answers to the questionnaire items are to be provided in the column labelled support, either by simply marking an answer to indicate a restricted choice (i.e., Yes or No) or by entering a value or a set or a range of values in the supported range column. (Note that there are some items where two or more choices from a set of possible answers may apply. All relevant choices are to be marked in these cases.)

Each item is identified by an item number, which is given in the first column. The second column (labelled item description) contains the question to be answered. The remaining columns may be labeled: references, status, support, allowed value(s), supported value(s), or mnemonic. The reference column contains the reference or references to the appropriate static conformance requirements or other clauses in IEEE Std 802.21. The status column contains the status value [mandatory, optional, not applicable, or conditional (see M.4.6)] of the item. The answer to the item is to be entered in the support column by either entering Yes or No in the space provided or, if present, mark the appropriate tick box beside the appropriate answer. For items that contain a supported value(s) column, in addition to answering the support column, enter the value or values supported for the item in the space provided.

A supplier may also provide further information, categorized as either Additional Information or Exception Information. When present, each kind of further information is to be provided in a further subclause of items labeled A<*I*> or X<*I*>, respectively, for cross-referencing purposes, where <*I*> is any unambiguous identification for the item (e.g., simply a numeral). There are no other restrictions on its format or presentation.

A completed PICS proforma, including any Additional Information and Exception Information, is the PICS for the implementation in question.

NOTE—Where an implementation is capable of being configured in more than one way, a single PICS may be able to describe all such configurations. However, the supplier has the choice of providing more than one PICS, each covering some subset of the implementation’s capabilities, if this makes for easier and clearer presentation of the information.

### Additional information

Items of Additional Information allow a supplier to provide further information intended to assist in the interpretation of the PICS. It is not intended or expected that a large amount of information will be supplied, and a PICS can be considered complete without any such information. Examples of such Additional Information might be an outline of the ways in which an (single) implementation can be set up to operate in a variety of environments and configurations, or information about aspects of the implementation that are outside the scope of this standard but have a bearing upon the answers to some items.

References to items of Additional Information may be entered next to any answer in the questionnaire, and may be included in items of Exception Information.

### Exception information

It may happen occasionally that a supplier will wish to answer an item with mandatory status (after any conditions have been applied) in a way that conflicts with the indicated requirement. No preprinted answer will be found in the Support column for this. Instead, the supplier shall write the missing answer into the Support column, together with an X<*I*> reference to an item of Exception Information, and shall provide the appropriate rationale in the Exception Information item itself.

An implementation for which an Exception Information item is required in this way does not conform to this standard.

NOTE—A possible reason for the situation described above is that a defect in this standard has been reported, a correction for which is expected to change the requirement not met by the implementation.

### Conditional status

The PICS proforma contains a number of conditional items. These are items for which both the applicability of the item itself, and its status if it does apply, mandatory or optional, are dependent upon whether or not certain other items are supported.

In this PICS proforma conditional items are represented through the use of nesting item numbering and by using individual conditional items as indicated in the status column.

If the value of the predicate is true, the conditional item is applicable, and its status is given by S and the support column is to be completed in the usual way. Otherwise, the conditional item is not relevant and the N/A answer is to be marked.

A predicate is an mnemonic for an item in the PICS proforma. The value of the predicate is true if the item is marked as supported, and is false otherwise.

Each item referenced in a predicate is indicated by an asterisk in the item number column.

## Identification of the implementation

In the space provided in the following subclauses, provide all information that will uniquely identify both the supplier (or client of the test laboratory) and the implementation and the system on which it resides. Also, provide a person as a point of contact for any queries concerning the contents of the PICS.

### Implementation and the system

### Supplier or client of the test laboratory

Name:

Address:

**M.5.3 Contact**

Name:

Address:

## Identification of the protocol

|  |  |  |
| --- | --- | --- |
| **Item number** | **Identification of protocol specification** | **Support** |
| M.6.1 | IEEE Std 802.21-2008 |  |
|  |  |  |

## Identification of corrigenda to the protocol

|  |  |
| --- | --- |
| **Identification of corrigenda implemented** | |
| Specification | Corrigenda implemented |
| IEEE Std 802.21-2008 |  |

## PICS proforma tables

### Global statement of conformance

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item number** | **Item description** | **Status** | **Support** | **Mnemonic** |
| M.8. 1 | Have all mandatory capabilities been implemented? | M | Yes [ ] No [ ] | GBL1 |

Answering “No” to this question indicates non-conformance to the IEEE Std 802.21. Non-supported mandatory capabilities are to be identified in the implementation conformance statement, with an explanation of why the implementation is non-conforming.

### Roles

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item number** | **Item description** | **References** | **Status** | **Support** | **Mnemonic** |
| M.8.2.1 | Is MISF supported in a MN? | 5.4 | O.1 | Yes [ ] No [ ] | RL1 |
| M.8.2.2 | Is MISF supported in a network entity? | 5.4 | O.1 | Yes [ ] No [ ] | RL2 |

### Major capabilities

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item number** | **Item description** | **References** | **Status** | **Support** | **Mnemonic** |
| \*M.8.3.1 | Is the Event Service (ES) supported? | 6.3 | O.2 | Yes [ ] No [ ] | MC1 |
| \*M.8.3.2 | Is the Command Service (CS) supported? | 6.4 | O.2 | Yes [ ] No [ ] | MC2 |
| \*M.8.3.3 | Is the Information Service | 6.5 | O.2 | Yes [ ] No [ ] | MC3 |
|  | (IS) supported? |  |  |  |  |
| M.8.3.3. | Is the TLV query method | 6.5.6.1, | MC3:O.3 | Yes [ ] No [ ] N/A [ ] | MC3.1 |
| 1 | supported? | 6.5.6.2 |  |  |  |
| M.8.3.3. | Is the RDF query method | 6.5.6.1, | MC3:O.3 | Yes [ ] No [ ] N/A [ ] | MC3.2 |
| 2 | supported? | 6.5.6.3 |  |  |  |
| M.8.3.3. | Is the push mode supported? | 5.3.4 | O | Yes [ ] No [ ] | MC3.3 |
| 3 |  |  |  |  |  |
| \*M.8.3.4 | Is capability discovery supported? | 6.2.3 | M | Yes [ ] No [ ] | MC4 |
| M.8.3.4. | Is Unsolicited Capability | 8.2.4.3.3 | O | Yes [ ] No [ ] | MC4.1 |
| 1 | Discovery supported? |  |  |  |  |
| M.8.3.4. | Is Solicited Capability | 8.2.4.3.4 | M | Yes [ ] No [ ] | MC4.2 |
| 2 | Discovery supported? |  |  |  |  |
| \*M.8.3.5 | Is the Registration Service supported? | 6.2.4 | MC2 OR MC3.3: M | Yes [ ] No [ ] | MC5 |
| M.8.3.6 | Is Mobile Initiated Handover supported? | 6.4.3.2.3 | O.4 | Yes [ ] No [ ] | MC6 |
| M.8.3.7 | Is Network Initiated | 6.4.3.2.4 | O.4 | Yes [ ] No [ ] | MC7 |
|  | Handover supported? |  |  |  |  |
| M.8.3.8 | Is MIS Acknowledgement protocol supported? | 8.2.2 | O | Yes [ ] No [ ] | MC8 |
| M.8.3.9 | Is MIS fragmentation supported? | 8.4.2 | M | Yes [ ] No [ ] | MC9 |

### PDUs

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item number** | **Item description** | **References** | **Status** | **Support** | **Mnemonic** |
| M.8.4. 1 | MIS\_Link\_Detected indication? | 8.6.2.1 | MC1 :M | Yes [ ] No [ ] N/A[ ] | PDU1 |
| M.8.4.2 | MIS\_Link\_Up indication? | 8.6.2.2 | MC1:M | Yes [ ] No [ ] N/A[ ] | PDU2 |
| M.8.4.3 | MIS\_Link\_Down indication? | 8.6.2.3 | MC1 :M | Yes [ ] No [ ] N/A[ ] | PDU3 |
| M.8.4.4 | MIS\_Link\_Going\_Down indication? | 8.6.2.5 | MC1:M | Yes [ ] No [ ] N/A[ ] | PDU4 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item number** | **Item description** | **References** | **Status** | **Support** | **Mnemonic** |
| M.8.4.5 | MIS\_Link\_Parameters\_Report indication? | 8.6.2.4 | MC1:M | Yes [ ] No [ ] N/A[ ] | PDU5 |
| M.8.4.6 | MIS\_Link\_Handover\_Imminent indication? | 8.6.2.6 | MC1:O | Yes [ ] No [ ] N/A[ ] | PDU6 |
| M.8.4.7 | MIS\_Link\_Handover\_Complete indication? | 8.6.2.7 | MC1 :O | Yes [ ] No [ ] N/A[ ] | PDU7 |
| M.8.4.8 | MIS\_Link\_Get\_Parameters | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU8 |
|  | request? | 8.6.3.1 |  |  |  |
| M.8.4.9 | MIS\_Link\_Get\_Parameters | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU9 |
|  | response? | 8.6.3.2 |  |  |  |
| M.8.4.10 | MIS\_Link\_Configure\_Threshold | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU10 |
|  | s request? | 8.6.3.3 |  |  |  |
| M.8.4.11 | MIS\_Link\_Configure\_Threshold | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU11 |
|  | s response? | 8.6.3.4 |  |  |  |
| M.8.4.12 | MIS\_Link\_Action request? | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU12 |
|  |  | 8.6.3.5 |  |  |  |
| M.8.4.13 | MIS\_Link\_Action response? | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU13 |
|  |  | 8.6.3.6 |  |  |  |
| M.8.4.14 | MIS\_Net\_HO\_Candidate\_Query | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU14 |
|  | request? | 8.6.3.7 |  |  |  |
| M.8.4.15 | MIS\_Net\_HO\_Candidate\_Query | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU15 |
|  | response? | 8.6.3.8 |  |  |  |
| M.8.4.16 | MIS\_MN\_HO\_Candidate\_Query | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU16 |
|  | request? | 8.6.3.9 |  |  |  |
| M.8.4.17 | MIS\_MN\_HO\_Candidate\_Query | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU17 |
|  | response? | 8.6.3.10 |  |  |  |
| M.8.4.18 | MIS\_N2N\_HO\_Query\_Resource | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU18 |
|  | s request? | 8.6.3.11 |  |  |  |
| M.8.4.19 | MIS\_N2N\_HO\_Query\_Resource | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU19 |
|  | s response? | 8.6.3.12 |  |  |  |
| M.8.4.20 | MIS\_MN\_HO\_Commit request? | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU20 |
|  |  | 8.6.3.13 |  |  |  |
| M.8.4.21 | MIS\_MN\_HO\_Commit | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU21 |
|  | response? | 8.6.3. 14 |  |  |  |
| M.8.4.22 | MIS\_Net\_HO\_Commit request? | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU22 |
|  | 8.6.3.15 |  |  |  |
| M.8.4.23 | MIS\_Net\_HO\_Commit | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU23 |
|  | response? | 8.6.3.16 |  |  |  |
| M.8.4.24 | MIS\_N2N\_HO\_Commit request? | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU24 |
|  |  | 8.6.3.17 |  |  |  |
| M.8.4.25 | MIS\_N2N\_HO\_Commit | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU25 |
|  | response? | 8.6.3.18 |  |  |  |
| M.8.4.26 | MIS\_MN\_HO\_Complete | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU26 |
|  | request? | 8.6.3.19 |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item number** | **Item description** | **References** | **Status** | **Support** | **Mnemonic** |
| M.8.4.27 | MIS\_MN\_HO\_Complete | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU27 |
|  | response? | 8.6.3.20 |  |  |  |
| M.8.4.28 | MIS\_N2N\_HO\_Complete | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU28 |
|  | request? | 8.6.3.21 |  |  |  |
| M.8.4.29 | MIS\_N2N\_HO\_Complete | 5.3.3.1, | MC2:M | Yes [ ] No [ ] N/A[ ] | PDU29 |
|  | response? | 8.6.3.22 |  |  |  |
| M.8.4.30 | MIS\_Get\_Information request? | 8.6.4.1 | MC3 :M | Yes [ ] No [ ] N/A[ ] | PDU30 |
| \*M.8.4.31 | MIS\_Get\_Information response? | 8.6.4.2 | MC3 :M | Yes [ ] No [ ] N/A[ ] | PDU3 1 |
| M.8.4.32 | MIS\_Push\_Information indication | 8.6.4.3 | MC3:M | Yes [ ] No [ ] N/A[ ] | PDU32 |
| M.8.4.33 | MIS\_Capability\_Discover request? | 8.6.1.1 | MC4:M | Yes [ ] No [ ] N/A[ ] | PDU33 |
| M.8.4.34 | MIS\_Capability\_Discover response? | 8.6.1.2 | MC4:M | Yes [ ] No [ ] N/A[ ] | PDU34 |
| M.8.4.35 | MIS\_Register request? | 8.6.1.3 | MC5:M | Yes [ ] No [ ] N/A[ ] | PDU35 |
| M.8.4.36 | MIS\_Register response? | 8.6.1.4 | MC5:M | Yes [ ] No [ ] N/A[ ] | PDU36 |
| M.8.4.37 | MIS\_DeRegister request? | 8.6.1.5 | MC5:M | Yes [ ] No [ ] N/A[ ] | PDU37 |
| M.8.4.38 | MIS\_DeRegister response? | 8.6.1.6 | MC5:M | Yes [ ] No [ ] N/A[ ] | PDU38 |
| M.8.4.39 | MIS\_Event\_Subscribe request? | 8.6.1.7 | M | Yes [ ] No [ ] | PDU39 |
| M.8.4.40 | MIS\_Event\_Subscribe response? | 8.6.1.8 | M | Yes [ ] No [ ] | PDU40 |
| M.8.4.41 | MIS\_Event\_Unsubscribe request? | 8.6.1.9 | M | Yes [ ] No [ ] | PDU41 |
| M.8.4.42 | MIS\_Event\_Unsubscribe response? | 8.6.1.10 | M | Yes [ ] No [ ] | PDU42 |

### PDU parameters

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item number** | **Item description** | **References** | **Status** | **Support** | **Supported value** |
| M.8.5.1 | Maximum supported MIS\_Get\_Information response message size (in octets)? | 8.6.4.2 | PDU31: M | Yes [ ] No [ ] N/A [ ] |  |

### Timers

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Item number** | **Item description** | **References** | **Status** | **Support** | **Allowed values** | **Supported values** |
| M.8.6.1 | TransactionLifeTime Timer | 8.2.3.5 | M | Yes [ ] No [ ] | 1–255 |  |
| M.8.6.2 | RetransmissionInterval Timer | 8.2.3.8.2 | M | Yes [ ] No [ ] | 1–255 |  |
| M.8.6.3 | ReassemblyTimer | 8.4.2.3 | M | Yes [ ] No [ ] | 1–255 |  |

# Authentication and key distribution procedures

**(informative)**

## MIS service access authentication

MN PoS MIS Service

Authentication

MISF MISF

Server

1. MIS\_Auth indication

2. MIS\_Auth request

3. MIS\_Auth response

4. Use AAA protocol to communicate with service authentication server.

5. Derive key hierarchy 5. Derive key hierarchy

6. MIS\_Auth request (AUTH)

7. MIS\_Auth response (AUTH)

Out of the scope of IEEE 802.21.

**Figure N.1—Mobile initiated access authentication phase**

MN PoS MIS Service

Authentication

MISF MISF

1. MIS\_Auth request

2. MIS\_Auth response

Server

3. Use AAA protocol to communicate with service authentication server.

4. Derive key hierarchy

4. Derive key hierarchy

5. MIS\_Auth request (AUTH)

6. MIS\_Auth response (AUTH)

Out of the scope of IEEE 802.21.

**Figure N.2—Network initiated access authentication phase**

## Push key distribution

MN Serving

PoA

Target

PoA

Serving PoS

MIS User

MISF

MAC

MISF

MIS User

1. MIS\_Push\_Key.request

2. MIS\_Push\_Key request

3. MIS\_Push\_Key.indication

4. PoS installs the media specific key to target PoA.

6. MIS\_Push\_Key response

5. MIS\_Push\_Key.response

7. MIS\_Push\_Key.confirm

8. MIS user installs the media specific key in

MAC layer

Out of the scope of IEEE 802.21.

**Figure N.3—Push key distribution**

## Proactive authentication



MN Serving

PoA

MSA/ Target

Serving PoS

MIS User

MISF

MAC

PoA

MISF

MIS User

1. MIS\_LL\_Auth.request

2. MIS\_LL\_Auth request

3. MIS\_LL\_Auth.indication

4. The LL frames are sent to MSA to execute proactive authentication.

5. The LL frames are obtained from

MSA to be sent to MN.

7. MIS\_LL\_Auth response

6. MIS\_LL\_Auth.response

8. MIS\_LL\_Auth.confirm

More rounds may be needed.

n. Install key to the MAC layer.

Out of the scope of IEEE 802.21.

**Figure N.4—Proactive authentication**

## Optimized pull key distribution



MN Serving

PoA

MSA/ Target

Serving PoS

MIS User

MISF

MAC

PoA

MISF

MIS User

AAA

1. MIS\_LL\_Auth.request

2. MIS\_LL\_Auth request

3. MIS\_LL\_Auth.indication

4. A key is installed to AAA.

5. The LL frames are sent to MSA.

6. The LL frames are obtained from MSA.

7. Contact AAA for MN authentication.

10. MIS\_LL\_Auth.confirm

9. MIS\_LL\_Auth response

8. MIS\_LL\_Auth.response

MN authentication with MSA (AAA) using MIS\_LL\_Auth.

n. Install key to the MAC layer.

Out of scope of IEEE802.21.

**Figure N.5—Optimized pull key distribution**

## Termination phase

MN PoS MISF MISF

1. MIS\_Auth\_Termination request

2. MIS\_Auth\_Termination response

**Figure N.6—MN initiated termination phase**

# Protection through transport protocols

(informative)

MIS messages can be carried over wireless protocols in layer 2 such as defined in IEEE Std 802.11 or layer

3 as defined in IETF RFC 5677. In the following, the security protection provided through the transport protocol are discussed and security issues are identified with each protection mechanism.

## Protection through layer 2

When MIS messages are transported over a layer 2 protocol, the protection may be provided through the layer 2 protocol such as TKIP and CCMP specified in IEEE Std 802.11.

The protection in layer 2 is usually established with L2 identifiers such as MAC address for an MN and a PoS. MIS messages are protected together with other data. Furthermore, if MIS messages are transported over different layer 2 protocols, then the protection may be different. If the PoS is not co-located with a PoA in the same device, the protection through a L2 protocol may not provide end-to-end security between the MN and the PoS.

On the other hand, such protection through a layer 2 protocol will not require any change on either MIS pro- tocol or the layer 2 protocol that transports the MIS protocol.

## Protection through IPsec

When MIS messages are transported over IP as defined in IETF RFC 5677, they may be protected by IPsec as specified in IETF RFC 4302 for IP Authentication Header (AH) and RFC 4303 IP Encapsulating Security Payload (ESP). When IPv6 is implemented in a MN and a PoS, then IPsec is mandatory. In this case, each MIS message is protected at IP layer as an IP payload in each IPsec packet.

For a pair of IP nodes with fixed IP addresses, the IPsec Security Associations (SAs) are established through Internet Key Exchange (IKEv1 or IKEv2) specified in IETF RFC 2409 and IETF RFC 4306. However, in case of MIS message protection, the IP address of a MN may be dynamic. In this case, a protocol suite defined by IETF RFC 4555 - “IKEv2 Mobility and Multihoming Protocol (MOBIKE)” may be used to establish SAs between an MN and a PoS (a.k.a. MoS as defined in IETF RFC 5677).

Similar to IKEv2, MOBIKE is a heavy weight protocol. The MOBIKE RFC is explicitly defined for tun- nel-mode IPSec connections.

IPsec protocols are well defined and can provide proper protection for its IP payload. When SAs are established between an MN and a PoS, they provide end-to-end protection. Using IPsec will not require any changes to either MIS protocol or IPsec.

Similar to protection provided in layer 2, the protection through IPsec are not MIS specific. However, for the mutual authentication through MOBIKE, the certificates may be issued on identifiers that are related to MIS applications. From this point of view, IPsec is closer to MIS specific protection, compared to L2 protection.

1. The Institute of Electrical and Electronics Engineers, Inc.

   3 Park Avenue, New York, NY 10016-5997, USA

   Copyright © 2013 by The Institute of Electrical and Electronics Engineers, Inc.

   All rights reserved. Published . Printed in the United States of America.

   IEEE is a registered trademark in the U.S. Patent & Trademark Office, owned by The Institute of Electrical and Electronics   
   Engineers, Incorporated.

   PDF: ISBN

   Print: ISBN

   IEEE prohibits discrimination, harassment, and bullying.

   For more information, visit <http://www.ieee.org/web/aboutus/whatis/policies/p9-26.html>.

   No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher. [↑](#footnote-ref-1)
2. Information on references can be found in Clause 2. [↑](#footnote-ref-2)
3. ANSI publications are available from the Sales Department, American National Standards Institute, 11 West 42nd Street, 13th Floor, New York, NY 10036, USA (http://www.ansi.org/). [↑](#footnote-ref-3)
4. IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, Piscataway, NJ 08854, USA (http://standards.ieee.org/). [↑](#footnote-ref-4)
5. The IEEE standards or products referred to in this clause are trademarks of the Institute of Electrical and Electronics Engineers, Inc. [↑](#footnote-ref-5)
6. Numbers preceded by P are IEEE authorized standards projects that were not approved by the IEEE-SA Standards Board at the time this publication went to press. For information about obtaining drafts, contact the IEEE. [↑](#footnote-ref-6)
7. IETF RFCs are available from the Internet Engineering Task Force website at http://www.ietf.org/rfc.html. [↑](#footnote-ref-7)
8. ISO publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembé, CH-1211, Genève 20, Switzer¬land/Suisse (http://www.iso.ch/). ISO publications are also available in the United States from the Sales Department, American National Standards Institute, 11 West 42nd Street, 13th Floor, New York, NY 10036, USA (http://www.ansi.org/). [↑](#footnote-ref-8)
9. ITU-T publications are available from the International Telecommunications Union, Place des Nations, CH-121 1, Geneva 20, Switzer¬land/Suisse (http://www.itu.int/). [↑](#footnote-ref-9)
10. W3C recommendations are available from http://www.w3.org. [↑](#footnote-ref-10)
11. IEEE Standards Dictionary Online subscription is available at:

    <http://www.ieee.org/portal/innovate/products/standard/standards_dictionary.html>. [↑](#footnote-ref-11)
12. [↑](#footnote-ref-12)
13. The mechanism that triggers and executes a link-layer handover/switch (also referred as an L2 handover) is specified within the corresponding media-specific standard and out of scope of this standard. Notes in text, tables, and figures are given for information only and do not contain requirements needed to implement the standard. [↑](#footnote-ref-13)