MBA 文書 0601 号

MBA Document 0601

The English translation Of MBA Standard 0201

MIS Protocol (MISP) Specification Ver. 1.02

The authoritive specification is Japansese one, MBA Standard 0201 (April 2004).

The Protocol Working Group in the Mobile Broadband Association (MBA) reviews a draft proposed by a member of the working group. After reviewing, the working group releases a proposal as a MBA standard through procedures.
This MBA standard 0201 was proposed by Mobile Internet Services, Inc. as `MIS Protocol (MISP) Specification' and released through procedures.

NOTE: This MBA standard does not mention industry property rights, which is mandatory in the standard. The right holders state, "the Furukawa Electric Co., Ltd. and Masataka OHTA own the rights, `Method to Share a Session Key, Wireless Terminal Authentication, Wireless Terminal and Wireless Base Router'. The licenses are available to any party on reasonable, nondiscriminatory, nonexclusive term for use except for those who partly or fully own the rights used in the standard and claim them ".

モバイルブロードバンド協会

Mobile Broadband Association

www.mbassoc.org

Revision History

2002/2/20	Fixed the length of Geographic Information Object from 12 to 14.
2002/5/31	Fixed the length of UpLink Type Object from 6 to 8.

Table of Contents

1. Introduction

MIS Protocol (MISP) is designed in order to connect a base router and a terminal.

MISP operates at a lower layer of network layer protocols such as IPv4 and IPv6. MISP can handle multiple upper layer protocols at the same time. Ethernet, IEEE802.11b and etc. can be used as a lower protocol of MISP Layer.

MISP has a mechanism so that a terminal by itself can discover a base router on the same medium. After a terminal connects to medium, it can discover a base router and establish a channel. MISP has an authentication system by user name and password. During authentication, a cryptographic mechanism is used to prevent a secret key from being stolen even if the channel is monitored on the medium. A base router and a terminal exchange a shared key and each packet between them can be encrypted and authenticated. Connection created by MISP looks like a Point-to-Point connection from the upper layer.

These functionalities are designed for wireless LAN environments and they are secure enough for wireless public services.

2. Terminology and Concepts

2.1. Account

Right to access a network by MISP.

2.2. Account Identifier

A byte stream in order to identify an account. The maximum length is 253 bytes.

An account Identifier is used as an NAI (Network Access Identifier) specified in RFC2486. Each account has a different Account Identifier.

2.3. Password

A byte stream in order to authenticate an account. The maximum length is 253 bytes.

Password is secret information per account.

2.4. MN

A mobile terminal. Abbreviation of a `Mobile Node`.

An MN freely moves around. Each MN has each account.

2.5. BR

Abbreviation of a `Base Router`.

A BR is fixedly installed and has a constant connection to the Internet. A BR acts as a router between an MN and the Internet. A BR also has a feature to validate the account of an MN.

2.6. AS

Abbreviation of an `Authentication Server'.

A BR can delegate the validation function of an account to an AS. Account and password data can be centralizedly managed at an AS by using a single mapping table of account identifier and password, for requests from multiple BRs.

2.7. Session

Communication between MN and BR.

Only one session can exist between each MN and each BR. Namely, one session does not effect other sessions.

2.8. Session Key

A key per session. A key is shared between MN and BR. There can be at maximum two session keys for one session. A session key is updated at the defined intervals.

2.9. Base Router Group

A group of BR's. A BR belongs to 0 or more of Base Router Groups.

A Base Router Group is expressed as 32-bit integer.

2.10. Security Type

A set of Authentication Type, Key Distribution Algorithm and Data Encryption Algorithm.

2.11. Message

Packets of MISP.

2.12. Medium

A method to exchange messages between MN and BR.

2.13. Channel

Communication path between MN and BR in order to exchange messages.

A medium can have one or more of channels. A medium specifies the number of channels.

3. A MISP Architecture

3.1. A System Architecture

MISP provides mutual authentication and etc. between MN and BR.

A MISAUTH server behind a BR helps the authenticate between MN and BR.

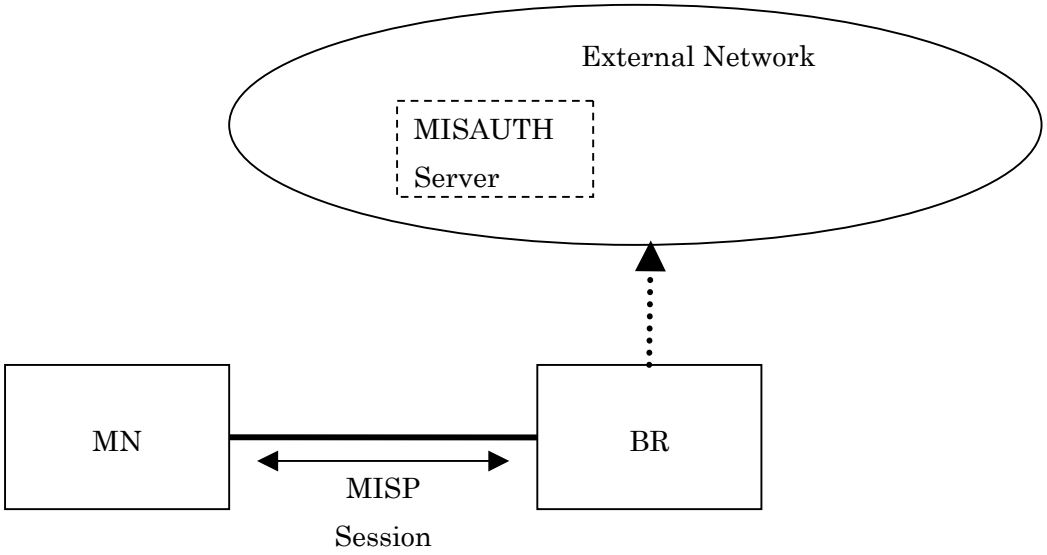

Fig. 1 A System Architecture

One BR can establish sessions with multiple MN's. On the other hand, one MN can have multiple sessions to different BR's. In either case, each session is independent of the other session.

A BR has a link to an external network. The upper layer of MISP is in charge of forwarding packets to an external network. MISP itself does not have a function to forward packets but to exchange packets between MN and BR.

3.2. Protocol Hierarchy

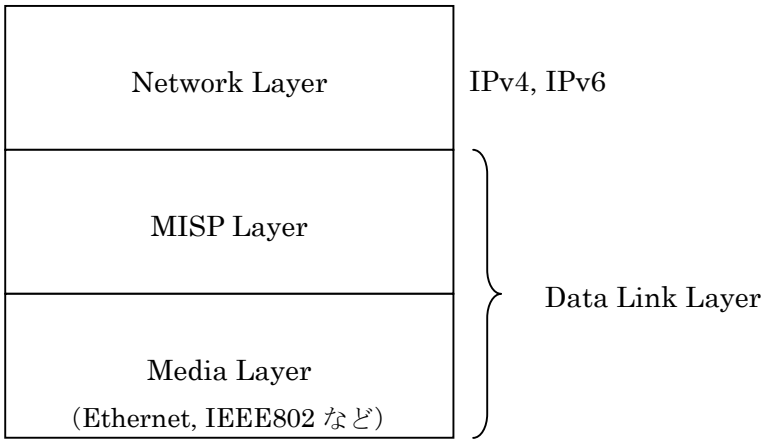

Fig. 2 The Protocol Hierarchy

In this specification, we call the lower layer of MISP layer a `network layer' and the upper layer a `medium layer'

This section describes requirements of the upper and lower layers

3.2.1. A Network Layer

The MISP layer provides the network layer with a packet forwarding function.

The MISP layer does not keep the precise length of each packet. The length of a received packet is never shorter but may be longer than the original length of the sent packet. If a received packet is longer than the sent packet, the original packet is put in the beginning of a received packet and one or more zeroes are padded.

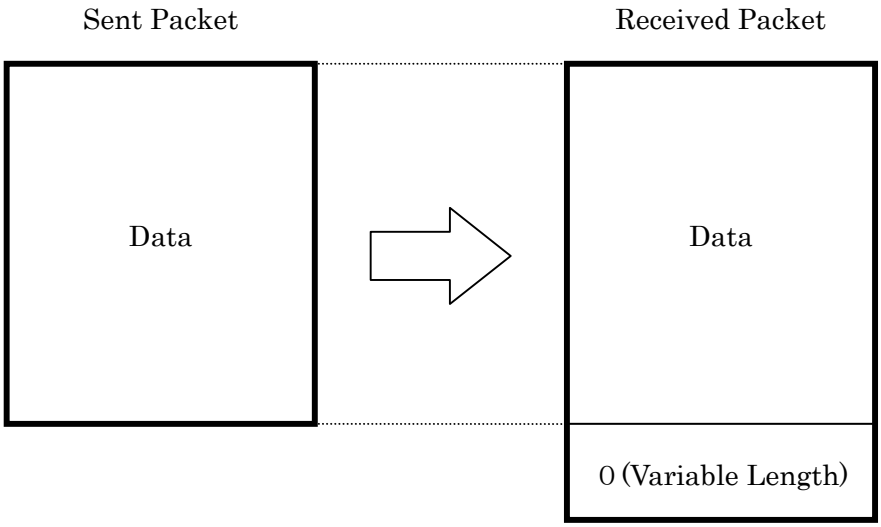

Fig. 3 Change of the Packet Length

Thus, the network layer must manage the length of packets if the precise length of the packet is necessary at the network layer.

3.2.2. The Medium Layer

The medium layer must have `MAC addresses'.

The media layer delivers a message from the MISP layer to an MN or a BR that has a MAC address specified by the MISP layer. The medium layer also deliver a received packet to the MISP layer, with the source MAC address of the received packet. In this case, the following conditions must be met:

- A message is delivered only to the MISP layer that has the same destination MAC address (the medium layer discards a message which has a different MAC address from that of the medium layer).
- Broadcast messages are delivered to all the MISP layer as possible.
- The same packets can be received more than once.
- The sent packets can be lost without any reception.

A MAC address is a byte stream of variable length from 1 byte to 128 byte. The

medium layer specifies the actual length of it.

A MAC address MUST meet the following conditions:

- MN and BR, which communicate with each other, MUST have a different MAC address.
- Each of MN's and BR's on one medium layer must have a different MAC address from all the MN;s and BR;s, when multiple connections resides on the medium layer.

MN and BR MUST use a global, unique MAC addresses.

3.3. Functions

MISP has functions as follows:

- Advertisement of information from a BR to an MN.
- MN authentication by a BR.
- BR authentication by an MN.
- Exchanges of session keys between BR and MN.
- Exchange of information for the network layer.
- Authentication and encryption of packets.

All the functions are described in details bellow.

3.3.1. Advertisement from a BR to an MN

A BR broadcasts beacon messages to MN's in the radio range. An MN knows information about BR's nearby by receiving the beacon messages.

3.3.2. MN Authentication by a BR

A BR validates an MN with a password associated with MN's account, which is included in an authentication request message.

3.3.3. BR Authentication by an MN

An MN validates a BR with information in an authentication success message from the BR.

3.3.4. Exchanges of Session Key between MN and BR

MN and BR know a shared session key information by exchanging authentication request and authentication success messages.

3.3.5. Exchanges of Information for Network Layer

MN and BR know information about a network layer by exchanging authentication request and authentication success messages.

3.3.6. Authentication and Encryption of Packets

MN and BR exchange network-layer packets and they authenticate or encrypt packets by the session key information.

3.4. Session

A `session' is relationship between MN and BR established by MISP.
Only one session can exist for each pair of MN and BR.
A session begins with an authentication success message.
A session finishes under the conditions as follows:

- When there is no valid session key.
- When a session termination message is sent.
- When an MN and/or a BR cannot communicate with one another physically.

A session is identified by a set of medium layer and MAC addresses of MN and BR.
A session has a channel.
A session has one security type, which never changes during the session.
A session has two session keys. One is `session key A', and the other is `session key B'.

4. Message Formats

4.1. Message Types

There are two types of messages:

- A data message
- A control message

A data message delivers network layer packets.
There are five types of control messages:

- Beacon
- Authentication Request
- Authentication Success
- Authentication Failure
- Session Termination

This section describes the detailed formats of the MISP header, objects and the details of the messages.

Note that all the messages are encoded in the network byte order (big-endian).

4.2. Message Structure

4.2.1. Message

A control message begins with the MISP header. The length of MISP header is 4 bytes. No message is shorter than 4 bytes. When receiving a message shorter than 4 bytes, it is discarded.

The type of a message is identified by the contents of MISP header. Zero or more numbers of `Objects' follow the MISP header. An Object is a set of type, length in byte and value.

4.2.2. A Data Message

A data message begins with the MISP header. The length of the MISP header is 4

bytes. No message is shorter than 4 bytes. When receiving a message shorter than 4 bytes, it is discarded.

The format following the MISP header is depend on the security type specified in a session.

4.3. A MISP Header

The length of a MISP header is 4 bytes and the structure is like this:

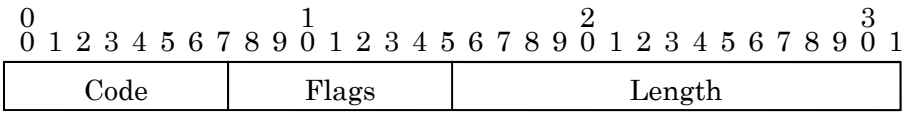

Fig. 4 MISP Header Format

Code Field

The code field specifies the type of this message. The length of the field is 1 byte and encoded in 8-bit integer. The values of the code can be:

0	Data
1	Beacon
3	Authentication Request
4	Authentication Success
8	Authentication Failure
9	Session Termination

A message including an unlisted code above must be ignored.

Flags Field

The flags field specifies additional information of a message. The length is 1 byte consisting of 8 bits. The meaning of each bit depends on the type of a message.

Length Field

The length field specifies the length of the whole message including the MISP header. The length of the length field is 2 bytes and encoded in unsigned 16-bit integer.
The tail of a message is ignored if the length of an actually-received message is longer than the length indicated by the length field. If the length of an actually-received message is shorter than the length indicated by the length field, the message must be ignored.

4.4. Object

The format of each object following the MISP header is as follows:

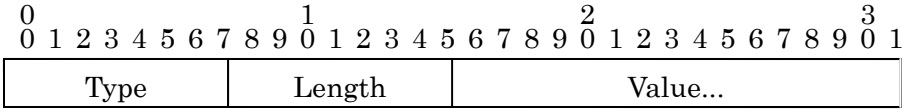

Fig. 5 Basic Format of Object

Type Field

The type field encoded in 8-bit integer specifies the type of the object. Types of an object defined in this specification are:

0	Padding
2	Beacon Time Stamp
3	IPv4 Local Address
4	IPv4 Remote Address
5	ICV (Integrity Check Value)
6	NAI (Network Access Identifier; see RFC2486)
8	Session Key Delivery Data
9	Geographic Information
10	Number of Available IPv4 Address Left
11	IPv4 Source Address Filtering

13	Error Reason
14	Base Router Group
15	Valid Period of Session Key
16	Serial Number
17	Beacon Interval
18	Security Type
19	Uplink Type
20	Channel
21	Network Layer

Length Field

The length field specifies the length of the whole object including type and length fields. The length field is 1 byte. The value of the length field includes the length of this field itself, so the minimum value is 2. If the value of the length field is less than 2 or the end of the value field is over the end of a message itself, the message must be ignored.
Note that a padding object does not have the length field.

Value Field

The value field specifies data of an object. It has variable length. The length of the value field is (*Length – 2)* bytes. If the length field specifies 2, the value field does not exist. The maximum length of this field is 253 bytes.
Note that a padding object does not have this field.

Bellow sections describe the format of each object. Though some of the format figures do not begin with 4-byte boundary, the alignment of objects is not limited. This is because easy understanding of the figures are taken into cosideration.

4.4.1. Padding Object

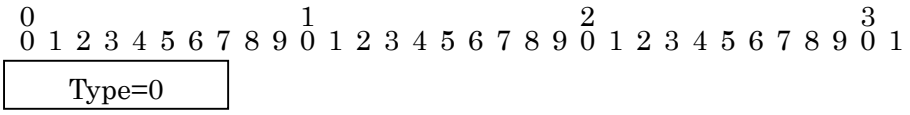

Fig. 6 Format of Padding Object

The meaning of each field is:

Type	0

A padding object does not carry any information and it is just ignored. If space is left between objects for any reason, any number of padding objects can be padded there.

The format of a padding object is special in terms of not having a length field.

4.4.2. Beacon Timestamp Object

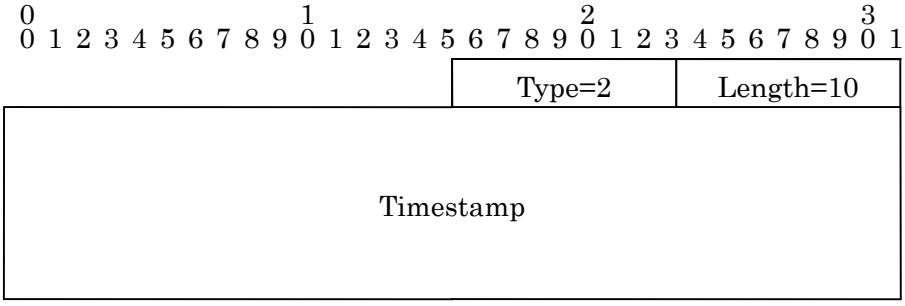

Fig. 7 Format of Beacon Timestamp Object

The value of each field is as follows:

Type	2
Length	10 (Fixed). The object is ignored if the length is not 10.
Timestamp	8 bytes, unsigned integer. The timestamp field specifies the

elapsed time sine 00:00:00 A.M., January 1st, 1970 GMT.

The value of the timestamp field is used as an identifier for the following authentication.

4.4.3. IPv4 Local Address Object

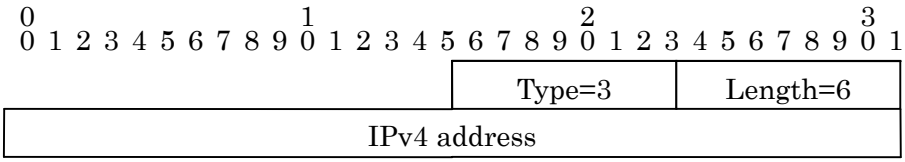

Fig. 8 Format of IPv4 Local Address Object

The value of each field is as follows:

Type	3
Length	6 (Fixed). This object is ignored if the length is not 6.
IPv4 address	An IPv4 address encoded in 4 bytes.

This object specifies an IPv4 address of a host sending the a including the object. For example, if a BR sends an authentication success message including this object to an MN, the object specifies the IPv4 address of the BR.

4.4.4. IPv4 Remote Address Object

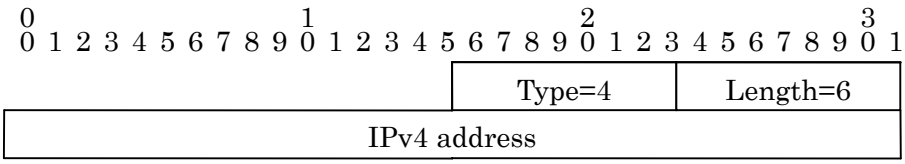

Fig. 9 Format of IPv4 Remote Address Object

The value of each field is as follows:

Type	4
Length	6 (fixed). This object is ignored if the length is not 6.
IPv4 address	An IPv4 address encoded in 4 byte.

This object specifies an IPv4 address of a host sending a message including the object.

For example, if a BR sends an authentication success message including this object to an MN, the object specifies the IPv4 address of the MN.

4.4.5. ICV Object

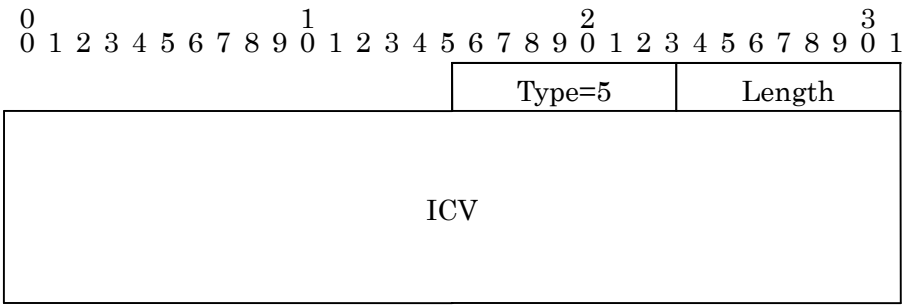

Fig. 10 Format of ICV Object

The value of each field is as follows:

Type	5
Length	Length of this object including the type and length fields in bytes.
ICV	A byte stream of *(Length – 2)* bytes.

This object specifies an ICV (Integrity Check Value) to validate the integrity of the whole message. The length and the meaning of ICV depend on each authentication method. Note that MN and BR share information about an authentication method by a security type object.

4.4.6. NAI Object

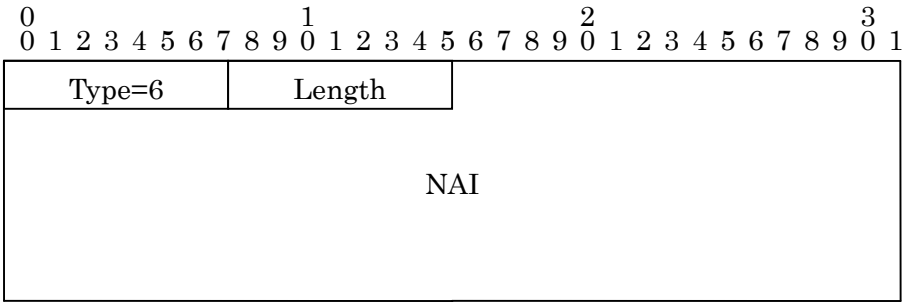

Fig. 11 Format of NAI Object

The value of each field is as follows:

Type	6
Length	The length of this object including the type and length fileds in bytes.
NAI	An account identifier. A byte stream of *(Length – 2)* bytes.

This object specifies an account identifier. MISP takes an identifier as a just byte stream. Note that a null character indicating a terminator must not be included. Also note that a null character at the end of an account identifier is taken as a part of the identifier.

4.4.7. Session Key Delivery Data Object

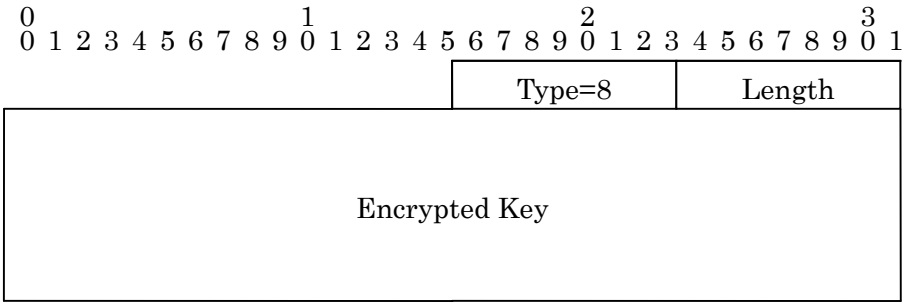

Fig. 12 Format of Session Key Delivery Data Object

The value of each field is as follows:

Type	8
Length	The length of this object including the type and length fields in bytes.
Encrypted Key	Data used to deliver a session key. A byte stream of *(Length – 2)* byte.

This object specifies data to deliver a session key between an MN and a BR. The data is usually encrypted to avoid interception of a key. The length and the meaning of data depend on each authentication method. Note that MN and BR share information of an authentication method by a security type object.

4.4.8. Geographic Information Object

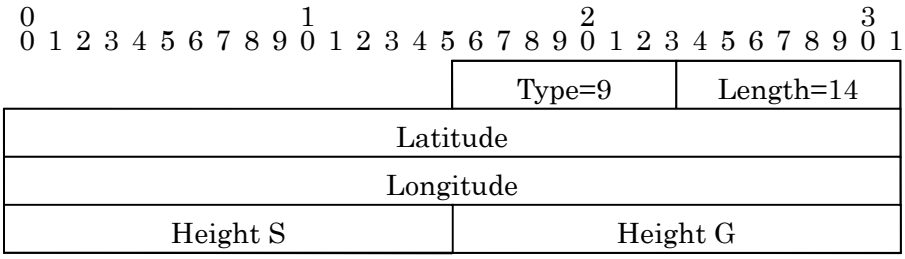

Fig. 13 Format of Geographic Information Object

The value of each filed is as follows:

Type	9
Length	14 (fixed). This object is ignored if Length is not 14.
Latitude	Latitude in signed 32-bit integer.
Longitude	Longitude in signed 32-bit integer.
Height S	Height above the sea level in signed 16-bit integer.
Height G	Height above the ground in signed 16-bit integer.

This object specifies that geographic location information by latitude, longitude and altitude.

Latitude and longitude are encoded in signed 32-bit integer respectively. The unit is 1/65536 degree. The sign of north latitude and east longitude are represented as positive, south latitude and west longitude as negative. Height above the sea level and ground are represented in meter unit and a positive value means high, negative value means being under the sea or the ground. Note that 0x80000000 means no information. The WGS84 geographic coordinate system is used as a geographic coordinate system.

4.4.9. Number of Available IPv4 Addresses Left Object

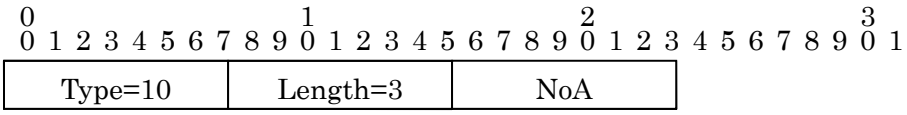

Fig. 14 Format of Number of Available IPv4 Addresses Left Object

The value of each field is as follows:

Type	10
Length	3 (fixed). This object is ignored if Length is not 3.
NoA	The number of available IPv4 addresses left. Encoded in unsigned number by 1 byte.

This object indicates the number of unassigned IPv4 addresses left in the IPv4 address pool at a BR. Because this number momentarily changes, all the IPv4 addresses can be run out, even if an MN sends an authentication request message to the BR right after having received a beacon message indicating that more than one IP addresses are left.

A BR may send this object whose NoA value is lower than the actual value.

Information of available IPv4 addresses left at a BR is not provided if this object is not sent.

4.4.10. IPv4 Packet Filter Object

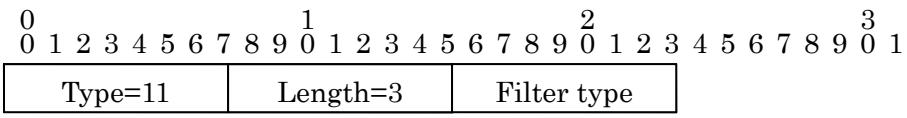

Fig 15 Format of IPv4 Packet Filter Object

The value of each field is as follows:

Type	11
Length	3 (fixed). This object is ignored if the Length is not 3.
Filter type	The type of a packet filter. Refer to the bellow table.

This object indicates whether or not IPv4 packets going through a BR, whch send this object, may be affected by a packet filter.

Filter types are:

0	There is no packet filter at a BR. All IPv4 packets are forwarded normally.
1	IPv4 packets may be affected by a packet filter. Only packets whose source or destination IPv4 address is specified by the IPv4 remote address object in the same message are allowed to go though the packet filter at the BR. In this case, it is required for an MN to use reverse tunneling, when employing Mobile IPv4.

This object is ignored if another value is set into the Filter Type. If this object does not exist, it indicates that IPv4 packets are not affected by a packet filter, which is the same as the Filter Type is set to 0.

4.4.11. Error Reason Object

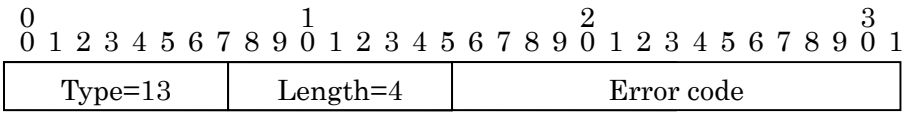

Fig 16 Format of Error Reason Object

The value of each field is as follows:

Type	13
Length	4 (fixed). This object is ignored if Length is not 4.
Error code	Indicate the error reason encoded in 16-bit integer.

When an error occurs, this object indicates the reason of the error. Error codes between 0 and 127 indicate a temporary error that may be recovered by an immediate retry. On the other hand, error codes between 128 and 255 mean permanent errors.

1	Could not communicate with an authentication server.
128	Authentication failure.
129	Lack of IPv4 addresses.
130	Invalid message format.

4.4.12. Base Router Group Object

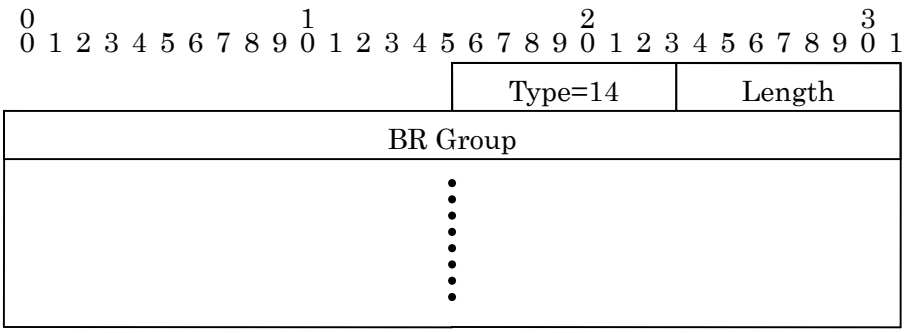

Fig. 17 Format of Base Router Group Object

The value of each field is as follows:

Type	14
Length	The length of this object including Type and Length fields. This value must be represented as 2+4n where n is an integer between 0 and 32. Otherwise, this object is ignored.
BR Group	A Base Router Group. Each Base Router Group is a 4-byte identifier; this object can contain more than 0 of base router groups.

This object indicates base router groups that the base router belongs to. This object can specify from 0 to 32 of base router groups.

If this object does not exist or if the value of the Length field is 2, it indicates that the BR belongs to no base router group.

4.4.13. Session Key Time to Live Object

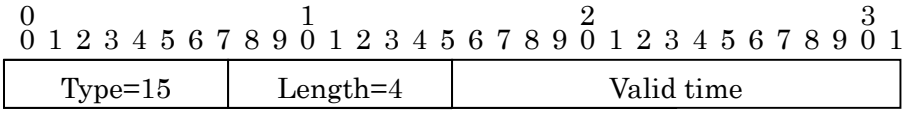

Fig. 18 Session Key Time to Live Object

The value of each field is as follows:

Type	15
Length	The length of this object including Type and Length field.
Valid time	Encoded in unsigned, 16-bit integer. The unit is second.

This object indicates the time to live of a session key.

4.4.14. Serial Number Object

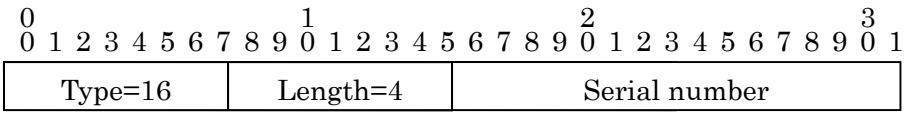

Fig. 19 Format of Serial Number Object

The value of each field is as follows:

Type	16
Length	The length of this object including Type and Length fields.
Serial number	Unsigned, 16-bit integer.

A serial number increases after this object is sent. The value returns to 0 after 0xffff. The initial number does not matter.

Serial numbers at receivers are not always sequential because of packet loss or duplicated packets.

A BR may send a serial number object in beacon messages. In this case, all the beacon messages from the BR must include serial number object. An MN can know lost beacon messages by checking the serial number objects in the received beacon messages.

4.4.15. Beacon Interval Object

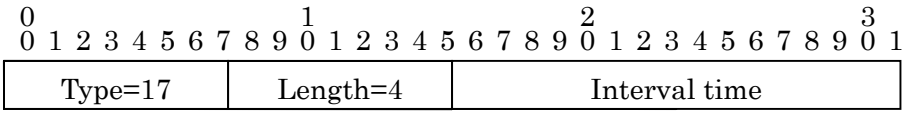

Fig. 20 Format of Beacon Interval Object

The value of each field is as follows:

Type	17
Length	The length of this object including Type and Length fields.
Interval time	Unsigned, 16-bit integer in millisecond.

This object indicates the interval of beacon message that a BR sends.

4.4.16. Security Type Object

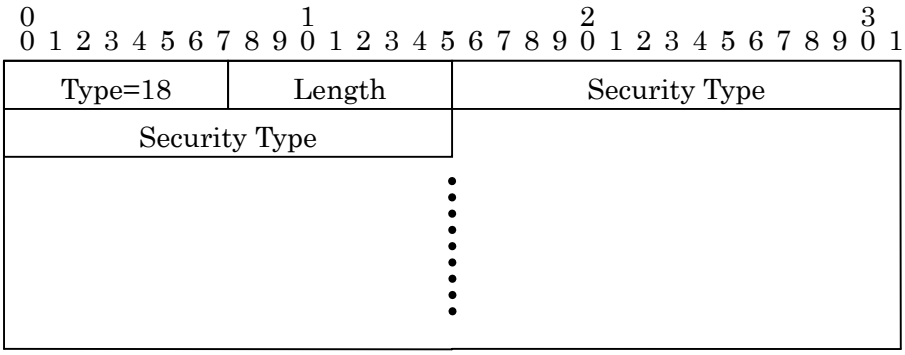

Fig. 21 Format of Security Type Object

The value of each field is as follows:

Type	18
Length	The length of this object including Type and Length fields. This value must be 2+2n where n is more than 0 and less than 127 integer. Otherwise, this object is ignored.
Security Type	Security type. More than 0 of security types can be listed.

This object indicates security types.

If this object is included in beacon messages sent by a BR, the security types supported by the BR are listed. More than 0 and less than 127 of security types can be specified.

If this object is sent by an MN in an authentication message, only one security type in this session must be specified. If multiple types are specified, an authentication will fail due to a format error.

4.4.17. Uplink Type Object

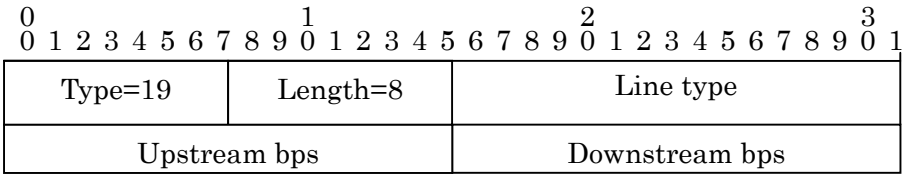

Fig. 22 Format of Uplink Type Object

The value of each field is as follows:

Type	19
Length	The length of this object. This value must be 8. Otherwise, this object is ignored.
Line type	Uplink types. Unsigned 16-bit integer. 0 FTTH (Optical Fiber) 1 xDSL 2 CATV
Upstream bps	Upstream bandwidth of the uplink. Unsigned, 16-bit integer.
Downstream bps	Downstream bandwidth of the uplink. Unsigned, 16-bit integer.

This object indicates the uplink bandwidth and type of a BR sending the object.

The bandwidth of the uplink is represented in unsigned, 16-bit integer and kbps. Note that 0xffff indicates that the bandwidth is more than 65.535 Mbps. In addition, note that the upstream means the direction from a BR to the external and the downstream means the direction from the external to a BR.

4.4.18. Channel Object

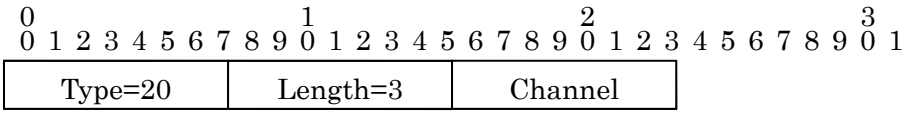

Fig. 23 Format of Channel Object

The value of each field is as follows:

Type	20
Length	The length of this object. This value must be 3. Otherwise, this object is ignored.
Channel	Unsigned, 8-bit integer.

This object indicates the channel that a BR uses. Representation of a channel depends on a medium.

If this object does not exist, it indicates that the medium has no channel switching function and has only one channel.

4.4.19. Network Layer Object

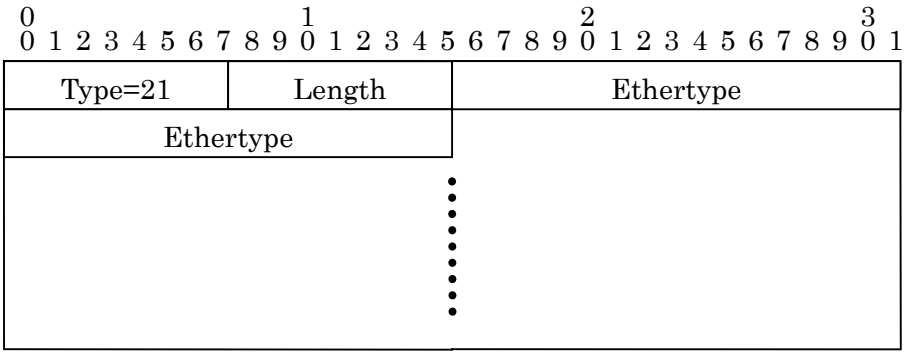

Fig. 24 Format of Network Layer Object

The value of each field is as follows:

Type	21
Length	The length of this object including Type and Length field. This value must be 2+2n where n is non-negative and less than 17 integer. Otherwise, this object is ignored.
Ethertype	Type of the network layer. In case of IPv4, this value is 0x0800. In case of IPv6, this value is 0x86dd.

If this object does not exist, no network layer is specified.

If a BR sends this object in beacon messages, this object indicates available network layers at the BR.

If an MN sends this object in an authentication message, this object indicates that the MN is requesting connection of the specified network layers. An MN can specify multiple network layers.

If a BR sends this object in an authentication success message, this object indicates available network layers in that session.

4.5. Message

4.5.1. Data Message

A data message conveys a packet of the network layer. Both of MN and BR send and receive data messages.

- Format

Format of data message is:

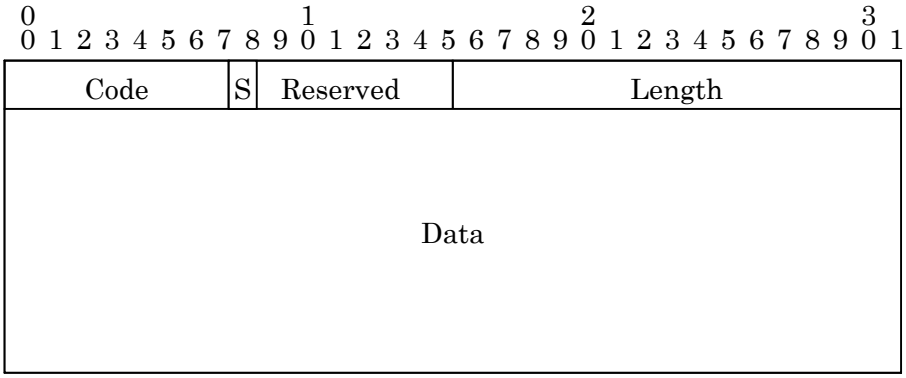

Fig. 25 Format of Data Message

The value of each field is as follows:

Code	0 (fixed).
S	This bit indicates the session key to be used for the authentication and decryption of this message. 0 indicates session key A, and 1 indicates session key B.
Reserved	0 (fixed).
Length	The whole length of this message in bytes. Unsigned, 16-bit integer.
Data	Format of this field depends on a security type. The length of this field is (Length - 4) byte.

S bit indicates the session key to be used in this data message.

The security type determines the format of the data field. Refer to `Security Type` sections for the format per security type.

4.5.2. Beacon Message

A BR periodically sends a beacon message. An MN can find available BR's from beacon messages.

- Format

The format of beacon message is:

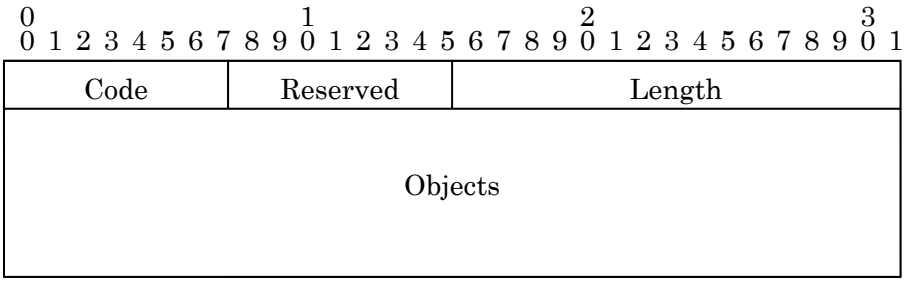

Fig. 26 Format of Beacon Message

The value of each field is as follows:

Code	1 (fixed).
Reserved	All bits are 0 (fixed).
Length	The whole length of this data message. Unsigned, 16-bit integer.
Objects	A List of required objects.

Mandatory objects which must be contained in a beacon message are:

Beacon Timestamp Object
Base Router Group Object
Serial Number Object
Beacon Interval Object
Security Type Object
Network Layer Object

Possible objects in a beacon message are:

Padding Object
IPv4 Available Addresses Left Object
IPv4 Packet Filter Object
Geographic Information Object

Uplink Type Object
Channel Object

If multiple same objects are contained, only first one is valid and the others are ignored.

Other objects not listed above are ignored.

➢ Sending

The value of timestamp field in beacon time stamp message from the same BR must monotonically increase; the value of the later sent beacon message must be greater than one of the former beacon messages.

The timestamp field specifies the elapsed time sine 00:00:00 A.M., January 1st, 1970 UTC.

➢ Receiving

The reserved field is not checked.

A beacon message that does not contain a beacon timestamp object is discarded.

4.5.3. Authentication Request Message

An MN sends an authentication request message to a BR. An authentication request message is sent under two conditions:

- When starting new session.
- When updating session key.

➢ Format

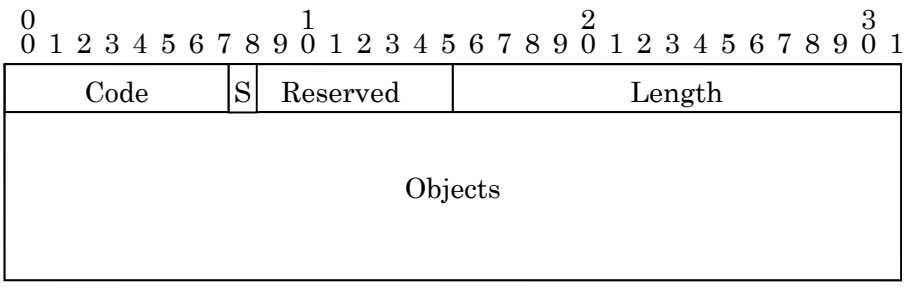

Fig. 27 Format of Authentication Request Message

The value of each field is as follows:

Code	3 (fixed).
S	This bit indicates the session key to be used for the authentication and decryption of this message. 0 indicates session key A, and 1 indicates session key B.
Reserved	All bits are 0 (fixed).
Length	The length of this message.
Objects	A list of required objects.

The mandatory objects that must be included in an authentication request message are:

Beacon Timestamp Object
Security Type Object
ICV Object
NAI Object
Session Key Delivery Data Object
Network Layer Object

Possible objects in an authentication request message are:

Padding Object
IPv4 Local Address Object

If multiple same objects are contained, only first one is valid and the others are ignored.

Other objects not listed above are ignored.

- Sending

The beacon timestamp object in this message contains the beacon timestamp in the beacon message associated with this authentication.

The Security type object in this message contains the security type requested in this authentication and the successive session.

S bit is always set to 0 when a new session is initiated.

➢ Receiving

The reserved field is not to be checked.

When one of the mandatory objects is missing, the beacon is discarded

4.5.4. Authentication Success Message

An authentication success message is sent from a BR to an MN.

➢ Format

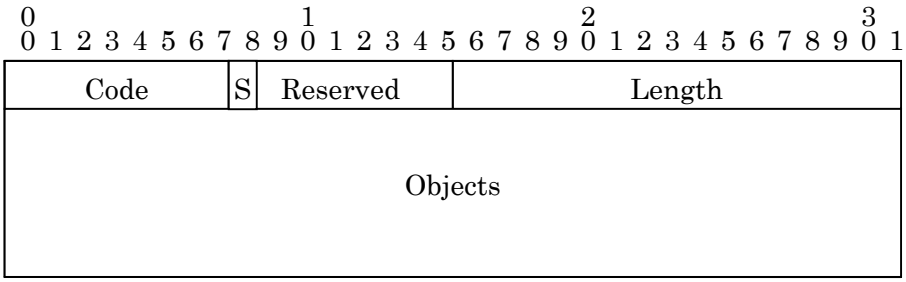

Fig. 28 Format of Authentication Message

The value of each field is as follows:

Code	4 (fixed).
S	This bit indicates the session key to be used for the authentication and decryption of this message. 0 indicates session key A, and 1 indicates session key B.
Reserved	All bits of this field are 0 (fixed).
Length	The length of this message.
Objects	A list of required objects.

The mandatory objects that must be included in an authentication success message are:

Beacon Timestamp Object
Session Key Time to Live Object
ICV Object
Network Layer Object

Possible objects in an authentication success message are:

Padding Objects
IPv4 Local Address Object
IPv4 Remote Address Object

If multiple same objects are contained, only first one is valid and the others are ignored.

Other objects not listed above are ignored.

- Sending

The beacon timestamp object in an authentication success message contains the beacon timestamp in the associated authentication request message.

The session key time to live object in an authentication success message contains time to live of the session key initiated from this message.

- Receiving

The reserved field is not checked.

If the mandatory objects are missing, the authentication success message is discarded and the authentication results in a permanent error.

4.5.5. Authentication Failure Message

An authentication failure message is sent from a BR to an MN. Note that an authentication failure message cannot be authenticated.

- Format

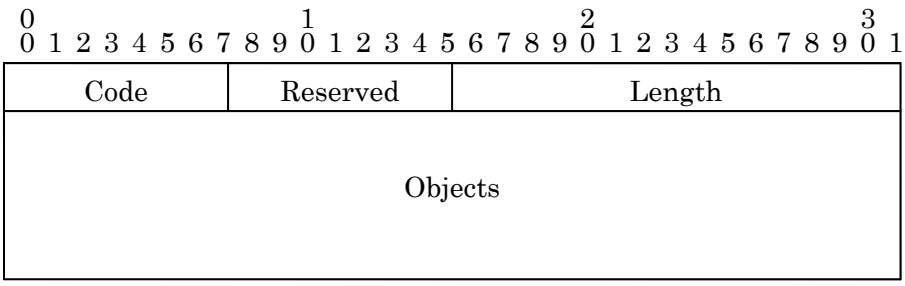

Fig. 29 Format of Authentication Failure Message

The value of each field is as follows:

Code	8 (fixed).
Reserved	All bits are 0 (fixed).
Length	The length of this message.
Objects	A list of required objects.

The mandatory objects that must be included in an authentication failure message are:

Beacon Timestamp Object
Error Reason Object

A possible object is:

Padding Object

If multiple same objects are contained, only first one is valid and the others are ignored.

Other objects not listed above are ignored.

- Sending

The beacon timestamp object in an authentication failure message contains the beacon timestamp in the associated authentication request message.

- Receiving

The Reserved field is not checked.

If the mandatory objects are missing, the authentication failure message is discarded.

4.5.6. Session Terminating Message

A session terminating message can be sent from both of MN and BR .

➢ Format

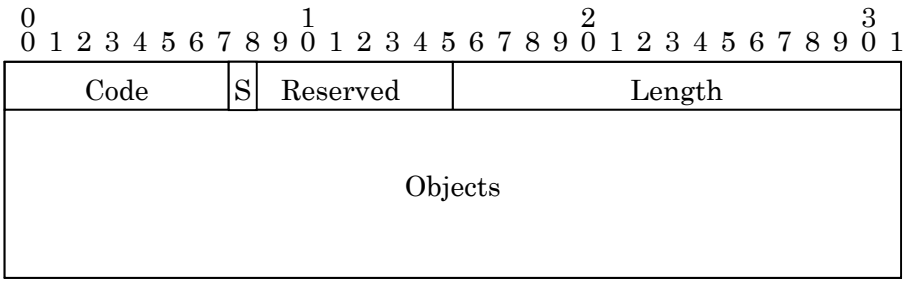

Fig. 30 Format of Session Terminating Message

The value of each field is as follows:

Code	9 (fixed).
S	This bit indicates the session key to be used for the authentication and decryption of this message. 0 indicates session key A, and 1 indicates session key B.
Reserved	All bits of this field are 0 (fixed).
Length	The length of this message.
Objects	A list of required objects.

The mandatory objects that must be included in an session terminating message are:

Beacon Timestamp Object
ICV Object

Possible objects in a session terminating message:

Padding Object
Error Reason Object

If multiple same objects are contained, only first one is valid and the others are ignored.

Other objects not listed above are ignored.

- Sending

The Beacon timestamp object in a session terminating message contains the beacon timestamp in the beacon message with which the session is initiated.

- Receiving

The reserved field is not checked.

If the mandatory objects are missing, the session terminating message is discarded.

5. Operation

5.1. Statically Configured Information

5.1.1. Information Configured to an MN

An MN is configured with a pair of an account identifier and a password.

5.1.2. Information Configured to a BR

A BR has a mapping table of account identifiers and passwords in advance.

In addition, an AS can be installed and connected to multiple BR's so that management of mapping table of account identifiers and passwords can be integrated to a single AS. This MISP specification does not define how AS and BR communicate with each other.

5.2. MN's Discovering/Selecting/Watching BR's

5.2.1. Sending Beacon Message (BR)

A BR periodically sends beacon messages on channels on which MISP is used. A BR must keep the sending interval as possible as it can be implemented. The value of the beacon interval depends on each medium. See Section of a medium.

Note that a beacon message is always sent without regard to the session between MN and BR.

5.2.2. Receiving and Watching Beacon Message (MN)

An MN receives packets on all the media configured on the MN and receives beacon messages. An MN tries to receive on all the channels of a medium, when it has several channels.

An MN makes a list of BR's around the MN. It depends on a medium how to make a list. See medium section.

5.2.3. Selecting BR (MN)

An MN selects a BR from a list of BR's based upon a certain criteria and tries to

initiate a session. An MN may have multiple sessions to multiple BR's if possible.

5.3. Initiating a Session

5.3.1. Receiving a Beacon Message (MN)

An MN initiates a session as follows:

1. Selects one beacon message from beacon messages sent by the BR that the MN tries to initiate a session with.
2. Determins a security type to be used in this session based upon the security type object in the beacon message. The MN selects a security type that is available on the BR from a list of security types in the security type object.
3. Creats a session key.
4. Creats an Authentication Request Message.
5. Operats an authentication process to an Authentication Request Message. The actual process depends on security type.
6. Sends the initial Authentication Request Message.
7. Re-sends the Authentication Request Message for 4 times after 100 ms, 300ms, 700ms and 1500 ms since sending the initial message. All the re-sent messages are the same as initial one.
   When an MN receives an authentication success message or authentication failure message which includes an associated beacon timestamp object or when it passes 3100ms after the initial authentication request message is sent, The MN quits this process and goes to the next step.
8. When the MN receives an Authentication Success Message or an Authentication Failure Message, the MN processes the message. If not, it means that session initiation with the BR is failed.

5.3.2. Receiving an Authentication Request Message (BR)

When a BR receives an Authentication Request Message from an MN with which the BR does not have session, the BR acts as follows:

1. The BR checks the beacon timestamp object in the received Authentication Request Message. If the value of the object is equal to or more than the

timestamp object in the last beacon message that the BR sent 5 seconds before, and less than the timestamp object in the last beacon message that the BR sent . If not, the BR discards this authentication Request Message, sends an authentication failure message to the MN.

2. The BR operates authentication and key delivery. The Details of this operation depends on security type. If the BR cannot communicate with an AS or if authentication is failed, the BR sends an authentication failure message to the MN, and then ends this process.
3. At this stage, the BR regards that a new session is successfully established and set an acquired session key as a session key A. Then, the BR invalidates session key B.
4. If an authentication request message has information about the network layer, the information is passed to the network layer. If the network layer has information to be informed in an authentication success message, the MIS layer accepts the information.
5. Forms an authentication success message.
6. Operats an authentication process for the authentication request message. The actual process depends on the security type.
7. Sends an authentication success message.

5.3.3. Receiving an Authentication Success Message (MN)

Receiving an authentication success message from a BR, an MN acts as follows:

1. Authenticates the message. The actual operation depends on a security type. If the authentication fails, an MN discards this message and the session initiation permanently fails.
2. At this stage, the session is successfully established. The MN set the acquired session key as a session key A. In addition, the MN sets a valid period of this session key in accordance with the session key time to live object. The MN invalidates session key B.
3. If the authentication success message has information about the network layer, the MIS layer passes the information to the appropriate network layer.

5.3.4. Receiving an Authentication Failure Message (MN)

After receiving an authentication failure message, an MN detects that the session initiation fails.

5.4. Updating a Session Key

When a newer session key between two keys expires within 10 seconds, an MN starts to update the key.

5.4.1. Receiving a Beacon Message (MN)

An MN that is going to update a session key acts at first as follows:

1. The MN receives beacon messages sent by the associated BR and selects one beacon message among them.
2. The MN creates a new session key.
3. The MN forms an authentication request message.
4. The MN operates authentication on the authentication request message. The actual operation depends on security type.
5. The MN sends the first authentication request message.
6. The MN sends four times, 100 ms, 300ms, 700ms, 1500ms after sending the first authentication request. These authentication request messages MUST be the same as first one.
   When receiving an authentication success message or an authentication failure message including the associated beacon timestamp object or when it passed 3100ms after sending the first authentication request, the MN gives up with this BR and goess to the next process.
7. If the MN receives an authentication success message or an authentication failure message, the MN moves to process that message. Otherwise, update of this session key results in failure.

5.4.2. Receiving an Authentication Request Message (BR)

When a BR that has established a session with an MN receives an authentication request message, the BR acts as follows:

(Note that the BR holds an old session key that is to be updated if the session key does

not expire yet even when the BR receives a new session key.)

1. The BR checks the beacon timestamp object in the received Authentication Request Message. If the value of the object is equal to or more than the timestamp object in the last beacon message that the BR sent 5 seconds before, and less than the timestamp object in the last beacon message that the BR sent If not, the BR discards this Authentication Request Message, send an authentication failure message to the MN.
2. The BR operates authentication and key delivery. The details of this operation depends on a security type. If the BR cannot communicate with an AS or if the authentication fails, the BR sends an authentication failure message to the MN, and then ends this process.
3. The BR sets the acquired session key as session key A or B specified in the authentication request message.
4. The BR forms an authentication success message.
5. The BR operates an authentication process on the authentication request message. The actual operation depends on a security type.
6. The BR sends the authentication success message to the MN.

5.4.3. Receiving an Authentication Success Message (MN)

Receiving an authentication success message, an MN acts as follows:

1. Authenticate the message. The actual operation depends on a security type. If this authentication fails, an MN discards this message and the session initiation permanently fails.
2. The MN regards that the session is established and sets the acquired session key to session key A or B in accordance with the S bit in the authentication success message. In addition, the MN set time to live of the session key in accordance with the session key time to live object.
3. If the authentication success message has information about the network layer, the MIS layer passes the information to the appropriate network layer.

5.4.4. Receiving an Authentication Failure Message (MN)

After receiving an authentication failure message, an MN detects that the session

initiation fails.

5.5. Exchanging a Data Message

Both of MN and BR can send and receive data messages.

5.5.1. Sending a Data Message

When the network layer passes packets to be sent to the MIS layer, the MIS layer acts as follows:

1. When there is no established session, the MIS layer returns an error to network layer and ends with the error.
2. The MIS layer checks whether the network layer is supported or not. If not, the MIS layer returns an error to the network layer and ends sending a data message.
3. The MIS layer selects a valid and newer key from two session keys on the session. The MIS layer forms a data message with adding authenticator and encryption. The actual operation of authentication and encryption depends on a security type.
4. The MIS layer sends a data message.

5.5.2. Receiving a Data Message

When receiving a data message, MIS layer acts as follows:

1. The MIS layer checks if the session key specified in the received data message is valid or not. If not, the MIS layer discards this message and end up this process.
2. The MIS layer authenticates and decrypts the data message. The actual processes depends on a security types. If the authentication fails, the MIS layer discards this message and ends this process.
3. The MIS layer checks if the network layer specified in the data message is supported or not. If not, the MIS layer discards this message and ends this process.
4. The MIS layer passes the decrypted packet to the network layer.

5.6. Terminating a Session

5.6.1. Active Termination of a Session

An MN or a BR can actively terminate a session when communication becomes unnecessary any more.

In this case, an MN or a BR acts as follows:

1. An MN or a BR selects a valid and newer session key from the session keys of the session. If there is no session key, an MN or a BR deletes the session information and ends this process.
2. An MN or a BR forms a session termination message.
3. An MN or a BR adds an authenticator in accordance with the security type.
4. An MN or a BR sends the session termination message.
5. An MN or a BR deletes the session information.

5.6.2. Receiving a Session Termination Message

When receiving a session termination message, an MN or a BR acts as follows:

1. An MN or a BR authenticates this message. The actual operation depends on a security type.
2. An MN or a BR deletes the session information.

5.6.3. Disappearing of a BR

An MN watches the existence of BR's. Normally, an MN watches them by receiving beacon messages. However, the method to watch the existence of BR's depends on media

If an MN detects disappearing of a BR, an MN handles the session with the BR as terminated.

5.6.4. Natural Extinction of a Session

When both of session key A and B expire, the session naturally terminates.

6. Security Types

Security types are as follows:
(Left number is used in the security type field in the security type object.)

1	Null
2	HMAC-MD5/HMAC-MD5/AES-CBC-128bit
3	HMAC-MD5/HMAC-MD5/HMAC-MD5-128bit

All the BR's and MN's MUST implement HMAC-MD5/HMAC-MD5/AES-CBC-128bit.
Other security types are optional.
The following sections describe each security type.

6.1. Null method

Null method uses no authentication and encryption.

6.1.1. Session Key

A session key is delivered. However, the key is not used for authentication. The length of a session key is 96bit (12byte).

6.1.2. An Authentication Request Message

- Sending (MN)

The ICV object is a byte stream whose length is any. The session key object includes a session key without any encryption.

- Receiving (BR)

A BR ignores the received ICV object with no check. Authentication never fails even though ICV object does not exist. If this message includes no session key delivery object, the BR uses 12-byte zeros as a session key.

6.1.3. An Authentication Success Message

- Sending (BR)

A BR puts a session key into the ICV object.

➢ Receiving (MN)

An MN ignores the received ICV object. Authentication never fails even though no ICV object is placed in the message.

6.1.4. Session Termination Message

➢ Sending

The ICV objects contains a session key.

➢ Receiving

The ICV object is ignored with no check. Authentication never fails even though no ICV object exists.

6.1.5. A Data Message

➢ Format

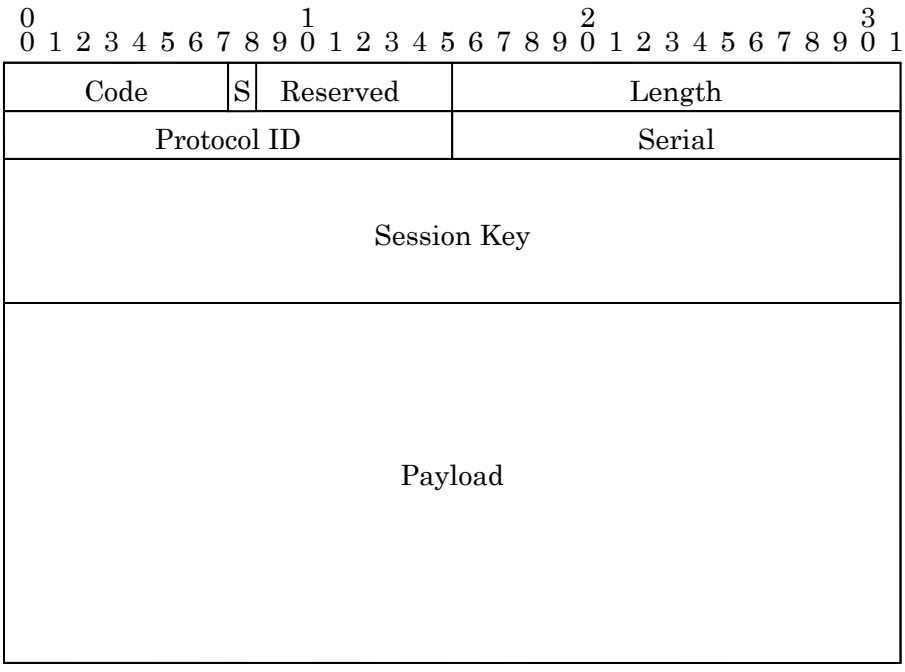

Fig. 1 Message Format of NULL Method

The value of each filed is as follows:

S	Specifying the session key used for an authentication. 1 bit value. 0 means session key A and 1 means session key B.
Reserved	All the bits of this field are 0 (fixed).
Length	The length of this message.
Objects	A list of required objects.
Protocol ID	The protocol ID of the upper layer protocol, whose length 2 bytes.
Serial	A serial number
Session Key	A session key whose length is 12 bytes.
Payload	Packet data of the upper protocol layer.

- MTU

The MTU of the network layer is the MTU of the medium layer minus 20 bytes.

- Sending

The Session Key field contains a session key. The Packet data of the upper layer protocol in the payload field is plain.

The Serial field contains a serial number. A serial number increases one by sending packets. Any initial serial number is acceptable. An MN or a BR manages a serial number independently. In addition, a serial number is independently managed for each session.

- Receiving

The Session Key field is not check and ignored. A BR or an MN accepts any data message without authentication.

6.2. The HMAC-MD5/HMAC-MD5/AES-CBC-128bit Method

MD5 and HMAC-MD5 are used for authentication, HMAC-MD5 is used to deliver session keys, and AES-CBC-128bit and MD5 are used to encrypt data.

6.2.1. A Session Key

The length of a Session Key is 128 bits (16 bytes).
A Session key is a 16-byte byte stream and is generated by applying HMAC-MD5 to seed of the session key. The seed of a session key is also a 16-byte byte stream and is generated by an MN for each session.
A method to generate a seed of a session key MUST meet the following conditions:

- It is hard to predict the next seed of a session key from the passed information
- A new seed of session key is different from any other seeds generated before.

Aseed may be appropriately calculated based upon timestamp of a beacon or the current time at an MN in order to meet these conditions.

6.2.2. An Authentication Request Message

➢ Sending (MN)

An MN puts a 16-byte byte stream into the ICV object and forms an authentication request message. The byte stream is generated as follows:

1. An MN forms an authentication request message containing an ICV object and the other required objects. A seed of a session key is set to the session key delivery object. 16-byte zeros are put into the value field of the ICV object. Note that the formed authentication request message includes the MISP header.
2. The MN creates a byte stream by concatenating source/destination MAC addresses and the authentication request message formed at stage 1.
3. The MN creates a 16-byte byte stream by applying MD5 to the byte stream formed at stage 2.
4. The MN creates a 16-byte byte stream by applying HMAC-MD5 with a password to the creates byte stream at stage 3.
5. The MN puts the 16-byte byte stream acquired at stage 4. onto the value field of ICV object formed at stage 1.

➢ Receiving (BR)

The ICV object contains a 16-byte byte stream. If the length of ICV object is different or if the ICV object does not exist, authentication fails.

The BR checks the 16-byte byte stream contained in the ICV object as follows:

1. The BR produces a 16-byte byte stream from the value field of the ICV object contained in the received authentication request message.
2. The BR replaces the value field in the ICV object with 16-byte zeroes.
3. The BR acquires a byte stream by concatenating the source/destination MAC addresses and the byte stream acquired at stage 2.
4. The BR acquires a 16-byte byte stream by applying MD5 to the byte stream acquired at stage 3.
5. The BR acquires a 16-byte byte stream by applying HMAC-MD5 to the byte stream acquired at stage 4.
6. The BR compares the byte stream acquired at stage 1. with one acquired at stage 5. If they are not the same, the authentication fails.

A session key delivery object contains a 16-byte byte stream. If the length of it is different or if a session key delivery object is not contained, authentication fails. The BR acquires a 16-byte byte stream as a session key, by applying HMAC-MD5 with the password to the 16 byte stream contained in the session key delivery object.

Authentication succeeds only when above all the checks are passed.

6.2.3. An Authentication Success Message

➢ Sending (BR)

A BR forms an ICV object in the same manner of an authentication request message, except for the point that HMAC-MD5 is applied with the session key instead of with the password.

➢ Receiving (MN)

MN checks an ICV object in the same manner of an authentication request message, except for the point that HMAC-MD5 is applied with the session key instead of with the password.

6.2.4. A Session Terminating Message

- Sending

An ICV object is generated in the same manner of a session success message.

- Receiving

An ICV object is checked in the same manner of a session success message.

6.2.5. A Data Message

- Format

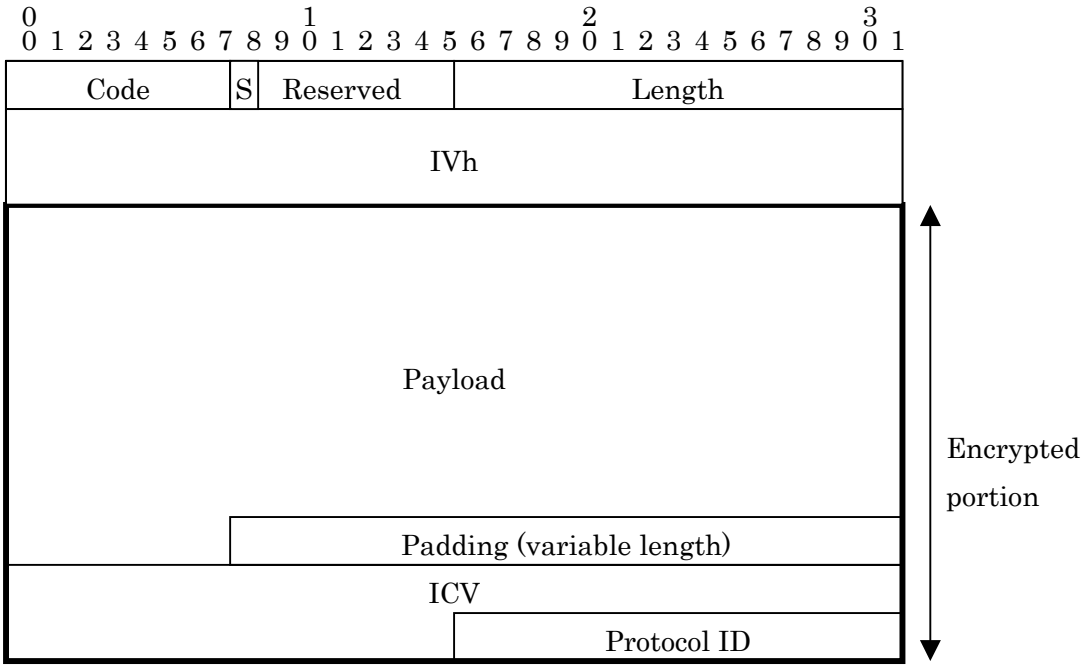

Fig. 6-2 Message Format of HMAC-MD5/HMAC-MD5/AES-CBC-128bit.

The value of each filed is as follows:

Code	0 (fixed).
S	Specifies a session key for encryption.
Reserved	All bits are set to 0.
Length	The length of the whole data message. The length MUST be 12 + 16n (n is an integer) in bytes. Otherwise, this message is

	ignored.
IVh	Heading 8 bytes of IV (Initialization Vector).
Payload	Packet data of the upper layer protocol. The length of this field is variable.
Padding	A byte stream of zeroes. The length of padding is between 0 byte and 15 byte.
ICV	A 6-byte byte stream. The Generation method is described later.
Protocol ID	An upper layer protocol ID of the payload. The length is 2 byte.

The encrypted portion in the above figure is encrypted by AES-CBC whose key length is 128 bits. The session key is used as a key of encryption. The MISP header and IVh are not encrypted. Because AES-CBC is block encryption and the block size is 16 byte, the length of padding is determined to make the length of encrypted portion multiple of 16. The length of padding is not held anywhere. Therefore, the length of original payload is unclear at the MIS layer. The upper layer protocol contained in the payload MUST manage the packet length by itself.

IVh MUST be generated by pseudo random number in order to reduce possibility that the same IVhs are generated. For example, IVh may generated by appropriately calculating with date and the previous IVh. The IV used for CBC is generated by IVh as follows:

1. Rotate 1 bit of each byte of IVh to left. This rotation is done byte by byte. These rotations result with 8-byte stream, IVl.
2. Generate a 16-byte byte stream, IV, consisting of heading 8 bytes of IVh and trailing 8-bytes IVl.

The ICV is heading 6 bytes of IVh.

- MTU

The network layer MTU is the medium MTU minus 20 bytes.

- Sending

A Data message is formed as follows:

1. Generate a 8-byte IVh.
2. Generate an IV from the IVh as described in the format section.

3. Form the MISP header.
4. Generate an ICV as described in the format section.
5. Generate encryption portion in accordance with the format, and encrypt the encryption portion.
6. Form the whole data message by adding the MISP header and the IVh.

➢ Receiving

A Data message is checked as follows:

1. Produce an IVh from a data message and generate an IV from the IVh.
2. Decrypt the encrypt portion.
3. Re-calculate an ICV and compare it with the ICV in the encrypt portion. If they are different, discard this data message.
4. Pass payload data to the network layer specified in the protocol ID. If the specified protocol ID is not supported, discard this data message.

6.3. HMAC-MD5/HMAC-MD5/HMAC-MD5-128 bit method

MD5 and HMAC-MD5 are used for authentication, HMAC-MD5 is used to deliver session keys and HMAC-MD5 is used to authenticate data messages.

6.3.1. A Session Key

The length of a session key is 128 bits (16 bytes).

Session key is a 16-byte byte stream generated by applying HMAC-MD5 with a password to a 16-byte seed of the session key. Note that the seed of the session key is generated by each MN.

A method of generating a seed of a session key MUST meet the following conditions:

- It is hard to predict the next seed of a session key from the passed information
- A new seed of session key is different from any other seeds generated before.

A seed may be appropriately calculated based upon timestamp of beacons or the current time at the MN in order to meet these conditions.

6.3.2. An Authentication Request Message

In receiving and Sending, emplay the same method as HMAC-MD5/HMAC-MD5/AES-CBC-128 bit method.

6.3.3. An Authentication Success Message

In receiving and Sending, emply the same method as HMAC-MD5/HMAC-MD5/AES-CBC-128 bit method.

6.3.4. A Session Terminating Message

In Receiving and Sending, emply the same methd as HMAC-MD5/HMAC-MD5/AES-CBC-128 bit method.

6.3.5. A Data Message

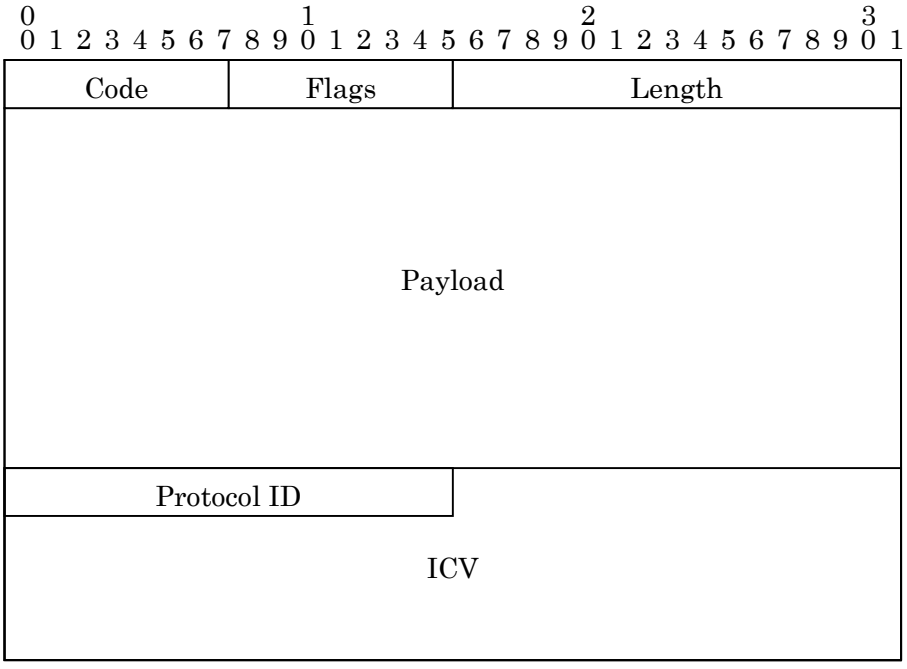

Fig. 6-3 Format of HMAC-MD5/HMAC-MD5/HMAC-MD5-128bit Method

The value of each field is as follows:

Code	0 for a data message.
Flags	0 (fixed).
Length	The total length of the whole data message in bytes.
Payload	Packet data of the upper layer protocol.
Protocol ID	An upper layer protocol ID of the payload. The length is 2 byte.
ICV	An ICV (14 bytes)

It is not necessary that the end of the payload fits 32-bit boundary even though the above figure shows as if it were. There is no space between the payload field and the protocol ID field.

An ICV is generated as follows:

1. Form a byte stream by concatenating the source/destination MAC addresses, the MISP header (Code, Flags and Length fields) and the protocol ID field. The length of byte stream is:
   (The length of source MAC address) + (The length of destination MAC address) + (The value of the length field) - 14
2. Acquire a 16-byte byte stream by applying HMAC-MD5 with the session key to the byte stream acquired at stage 1.
3. The heading 14-byte byte stream of the acquired stream at stage 2. is used as an ICV.

- MTU

The network layer MTU is the medium MTU minus 20.

- Sending

A data message is formed as follows:

1. Form all the fields of data message except for the ICV field.
2. Generate an ICV as described in the format section.
3. Form the whole data message by adding ICV field.

- Receiving

A data message is checked as follows:

1. Produce an ICV from the ICV field in the received data message.
2. Calculate an ICV as described in the format section.
3. Compare two byte streams acquired at stage 1. and 2. If they are different, discard this data message.
4. Pass the payload data to the network layer specified by the protocol ID. If the specified protocol ID is not supported, discard this data message.

7. Media

This section describes some issues depend on each medium supported by MISP.

7.1. Ethernet

7.1.1. MAC Address

A 6-byte Ethernet address is used as MAC address.

7.1.2. Format

When a MISP message is exchanged over Ethernet, 0x8893 is used as EtherType and a MISP message beginning with the MISP header is contained in a Ethernet payload.

7.1.3. A Beacon Message Sending Interval

A beacon message is sent for each 1 second.

7.1.4. MN's Watching a BR

When receiving a beacon message, an MN detects a BR sending that beacon and registers the BR into the BR list.

An MN removes a BR from the BR list when the MN receives no beacon from the BR for 3.5 seconds.

7.2. IEEE Std 802.11b

7.2.1. MAC Address

A 48-bit "Universal LAN MAC address" defined in IEEE Std 802-1990 is used as a MAC address.

7.2.2. Format

When a MISP message is exchanged over an IEEE802.11b link, a MISP message is encapsulated in accordance with the format defined in RFC1042. 0x8893 is used as EtherType.

7.2.3. The Beacon Message Sending Interval

The beacon message sending interval is 30 milliseconds.

7.2.4. MN's Behavior

An MN that has only one device to receive radio is recommended to act as follows:

- Channel Scan by an MN

An MN waits for beacon message for 100 milliseconds for each channel and builds up a list of BR's required to the MN in the signal strength order.

- Selecting a BR by an MN

An MN sends an authentication request message to a BR in the order listed in the BR list until authentication succeeds. When no session is established with any listed BR, the MN returns to start channel scan.

- Watching BR's by an MN

An MN regards the current BR as unavailable and returns to start channel scan when:

- The MN receives no beacon message from the BR for 300 milliseconds.
- The MN cannot receive 40 percent or more of beacon messages.
- For 30 milliseconds, the MN cannot receive any beacon message whose signal strength is more than the configured value.

8. Network Layers

This section describes some issues depend on each network layer supported by the MISP layer.

8.1. IPv4

8.1.1. The Protocol Number

The protocol number of IPv4 is 0x0800.

8.2. Dynamic IPv4 Address Allocation

- Function

A BR can inform IPv4 addresses that should be used at each end point of the Point-to-Point link established by MISP of an MN.

- A Beacon Message

A BR may send an IPv4 available address left object contained in a beacon message. When an MN receives this object and the value of this object is zero, that is, no IPv4 address is available; the MN does not initiate a session with the BR.

- An Authentication Request Message

An MN can indicate the assigned IPv4 address by put an IPv4 local address object into an authentication request message.

- An Authentication Success Message

A BR sends an authentication success message that contains respectively BR's IPv4 address in an IPv4 local address object and MN's IPv4 address in an IPv4 remote address object.

- Terminating a Session

When a session is terminated, the BR releases an IPv4 address used in the session.

- Cordination with Mobile IP

When a session is established, an IP address used in a session is changed, or a session is finished, an MN informs this event of the mobile IP module.

8.2.1. Notification of Existence of Packet Filter

A BR can inform possibility that a packet filter affects packets through the BR of an MN.

- An IPv4 Packet Filter Object

A BR provides information about a packet filter in an IPv4 packet filter object of a beacon message to MN's.

Currently, a BR can provide information about a packet filter of source IP address based filtering. When this type of packet filter exists, packets whose source IPv4 address is not specified in the IPv4 remote address object may not be forwarded correctly. This affects protocols over IPv4. For example, an MN MUST use reverse tunneling when using mobile IPv4 under this circumstance.

9. The Old Version of MIS Protocol

9.1. EtherType

0xaaaa is used as EtherType.

9.2. Beacon

A BR periodically sends a beacon message for each 30 milliseconds.

9.3. Security Types

There is no negotiation of security types. Authentication, key delivery and data communication are done as described bellow.

The number of security type, 1, is used during negotiation of the security type if this version of the security type is used over a newer protocol. Note that this SHOULD be provided only for test and backward compatibility and SHOULD NOT used for the actual use.

9.3.1. Authentication

Authentication is done by using an ICV object in an authentication request message.

An MN calculates an ICV by applying HMAC-MD5 with the password to the whole authentication request message when the MN sends the authentication request message. During this calculation, the length of the ICV object is 18 and the value filed of this object is set to 16-bit zeroes. The MN puts the result of this calculation to the value field in the ICV object, then the MN sends the authentication request message.

When A BR receives an authentication request message, the BR calculates an ICV same as an MN does. That is, A BR remembers a 16-byte byte stream of the value field in the ICV object and fills the value field with 16-byte zeroes. Then, the BR calculates an ICV by applying HMAC-MD5. The BR compares the result of the calculation and the original value in the ICV object. If both are the same, this indicates that the authentication request message is valid. Otherwise, the authentication request message is regarded as invalid.

9.3.2. Session Key Delivery

A session key is a 16-byte byte stream shared between an MN and a BR.

An MN generates a session key with a random number. The MN sends an encrypted session key contained in the session key delivery object in an authentication request message in order to deliver the session key to a BR. Encryption uses the inverse mapping of HMAC-MD5 with the password. Decryption uses HMAC-MD5 with the password.

Actually, the inverse mapping of HMAC-MD5 cannot be calculated, MN generates a session key as follows: An MN generates a 16-byte random number and uses it as an encrypted session key. Then, the MN decrypts the encrypted session key by applying HMAC-MD5 to it and uses the result as a session key.

9.3.3 Data Encryption

There is no data encryption. Only authentication per packet is done. Authentication uses HMAC-MD5.

Refer to later section, a Data Message, for details.

9.4. Objects

The old version of MISP uses only the following objects:

2	Timestamp
3	IPv4 local address
4	IPv4 remote address
5	ICV (Integrity Check Value)
6	NAI (Network Access Identifier; see RFC2486)
8	Session key delivery data

Other objects are ignored.

9.5. Messages

9.5.1. A Data Message

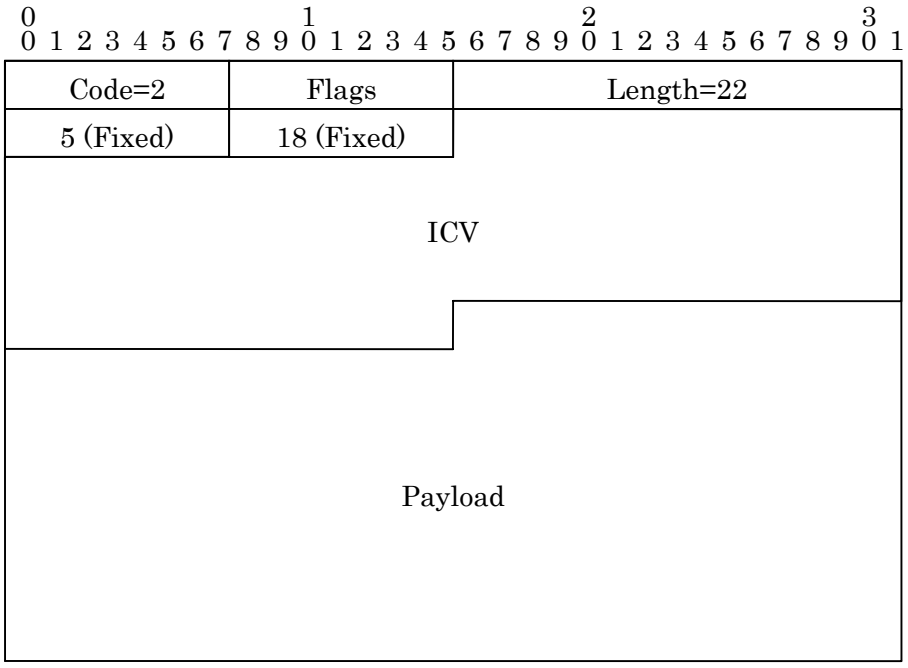

Fig. 9-1 Format of Data Message

The value of each field is as follows:

Code	2
Flags	0 (fixed). Other values are ignored.
Length	22 (Fixed). Operation is uncertain if other values are specified.
ICV	A Result of applying HMAC-MD5 with a shared key to the whole message. This field is set to 16-byte zeroes when the ICV is calculated.
Payload	IPv4 packet data

IPv4 is supported as the upper protocol layer. While the new MISP uses code 0 for

IPv4, the old MISP uses code 2.

9.5.2. An Authentication Success Message

In the old MISP, beacon timestamp object and session key time to live object are not sent. Received those objects are ignored. The session key time to live is fixedly 120 seconds.

9.5.3. An Authentication Failure Message

In the old MISP, an authentication failure message is never sent. Received this message is ignored.

9.5.4. An Session Terminating Message

In the old MISP, a session terminating message is never sent. Received this message is ignored.