



Question(s): 6/17

Geneva, 17 - 26 September 2014

Ref.: TD 1394 Rev.2**Source:** ITU-T Study Group 17**Title:** LS on new work item on simple encryption procedure for IoT device security

LIAISON STATEMENT**For action to:****For comment to:** ETSI TC ITS WG2, 3GPP TSG GERAN, ISO/IEC JTC 1/SC 27/WG 2, oneM2M**For information to:****Approval:** ITU-T Study Group 17 meeting (26 September 2014)**Deadline:** 31 March 2015**Contact:** Jonghyun Baek

Rapporteur of ITU-T SG17 Question 6/17

Tel: +82 2 405 6540

Fax: +82 2 405 5219

E-mail: jhbaek@kisa.or.kr

ITU-T Study Group 17, Security, coordinates security-related work across all ITU-T Study Groups. Often working in cooperation with other standards development organizations (SDOs) and various ICT industry consortia, SG17 deals with a broad range of standardization issues.

SG17 is pleased to inform you that we established a new work item in draft Recommendation ITU-T X.iossec-1, simple encryption procedure for IoT device security.

This work item is to provide an encryption procedure for Internet of things (IoT) device security. The procedure is intended to be applied to IoT devices that are devices with the mandatory capabilities of communication and the optional capabilities of sensing, actuation, data storage and data processing. This Recommendation provides specification of encryption with associated mask data (EAMD) for the IoT devices. It describes what EAMD does and how to provide a set of security services for traffic using it.

We kindly ask you to comment. Further, we would like to continue communication and cooperation with you.

Attachment: 2

- Annex 1 - A.1 justification for proposed draft new Recommendation X.iossec-1
- Annex 2 - draft Recommendation ITU-T X.iossec-1, Simple encryption procedure for IoT device security (X.iossec-1).

Attention: Some or all of the material attached to this liaison statement may be subject to ITU copyright. In such a case this will be indicated in the individual document.

Such a copyright does not prevent the use of the material for its intended purpose, but it prevents the reproduction of all or part of it in a publication without the authorization of ITU.

Annex 1 - A.1 justification for proposed draft new Recommendation X.iotsec-1

Question:	6/17	Proposed new ITU-T Recommendation	Geneva, 17-26 September 2014
Reference and title:	X.iotsec-1, Simple encryption procedure for IoT device security		
Base text:	Annex 2	Timing:	March 2016
Editor(s):	Hiroataka YOSHIDA, Hitachi, hirotaka.yoshida.qv@hitachi.com	Approval process:	TAP
<p>Scope (defines the intent or object of the Recommendation and the aspects covered, thereby indicating the limits of its applicability):</p> <p>This Recommendation is to provide an encryption procedure for the Internet of Things (IoT) device security. The procedure is intended to be applied to IoT devices that are devices with the mandatory capabilities of communication and the optional capabilities of sensing, actuation, data storage and data processing. This Recommendation provides specification of encryption with associated mask data for the IoT devices. The Recommendation includes what EAMD does and how to provide a set of security services for traffic using it.</p>			
<p>Summary (provides a brief overview of the purpose and contents of the Recommendation, thus permitting readers to judge its usefulness for their work):</p> <p>It is considered that the Internet of Things (IoT) is one of the most important areas for future standardization. From the ITU-T perspective, IoT is defined as a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies in [b-ITU-T Y.2060]. The IoT applications include various kinds of applications such as intelligent transportation systems (ITS), and smart home. The IoT reference model is composed of four layers, namely, the application layer, the service support and application support layer, the network layer, and the device layer in [b-ITU-T Y.2060]. The minimum requirement of the devices in the IoT is their support of communication capabilities. A high-level requirement which is relevant for the IoT is security. However, there has been no investigation of Recommendations on data confidentiality and integrity protection techniques that offer security for the device layer in IoT. Therefore, the device layer security is a missing area in IoT, hence this area should be studied and discussed for future standardization.</p>			

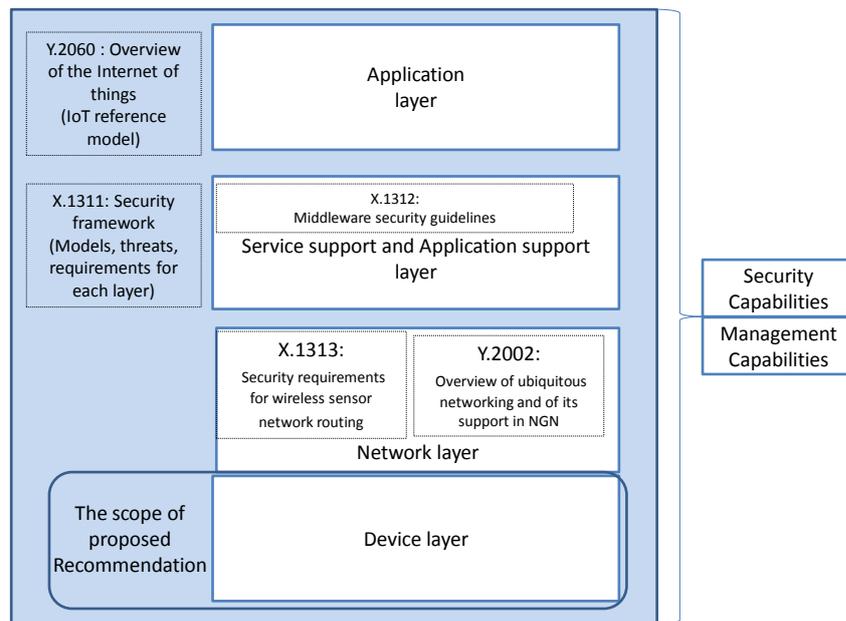


Figure 1 - The scope of X.iotsec-1 in the IoT reference model

This Recommendation provides specification of encryption with associated mask data (EAMD) for the Internet of Things (IoT) devices. EAMD only encrypts data, within a communication packet, whose security level is high. The associated mask data is used to indicate the security levels of data at each position within a packet. This draft includes what EAMD does and how to provide a set of security services for traffic using it.

Relations to ITU-T Recommendations or to other standards (approved or under development):

- [ITU-T F.744] Service description and requirements for ubiquitous sensor network middleware
- [ITU-T F.747.2] Deployment guidelines for ubiquitous sensor network applications and services for mitigating climate change
- [ITU-T X.1311] Information technology - Security framework for ubiquitous sensor networks
- [ITU-T X.1312] Ubiquitous sensor network middleware security guidelines
- [ITU-T X.1313] Security requirements for wireless sensor network routing
- [ITU-T X.usnsec-3] Secure routing mechanisms for WSN
- [ITU-T Y.2002] Overview of ubiquitous networking and of its support in NGN
- [ITU-T Y.2221] Requirements for support of ubiquitous sensor network (USN) applications and services in the NGN environment
- [ITU-T Y.2060] Next Generation Networks – Frameworks and functional architecture models
Overview of the Internet of things
- [ITU-T Y.2061] ITU-T Recommendation Y.2061 (2012), Requirements for support of machine oriented communication applications in the NGN environment

Liaisons with other study groups or with other standards bodies:

- SG13
- SG16
- IoT-GSI
- ISO/IEC SC27/WG2

- ETSI TC ITS WG2
- 3GPP TSG GERAN

Supporting members that are committing to contributing actively to the work item:

Brazil, China Unicom, Hitachi, Ltd., NICT, Uganda.

Annex 2

Draft Recommendation ITU-T X.iotsec-1

Simple encryption procedure for IoT device security (X.iotsec-1)

Summary

It is considered that the Internet of things (IoT) is one of the most important areas for future standardization. From the ITU-T perspective, IoT is defined as a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things.

In certain IoT devices, there is a real-time processing requirement that tasks are processed within a certain period of time. To ensure data confidentiality and integrity protection, one of the most basic countermeasures is the application of data encryption/authentication algorithms. The problem with the standard applications of data encryption/authentication algorithms is that this requirement could not be met.

This Recommendation provides specification of encryption with associated mask data (EAMD) for the Internet of things (IoT) devices. Recommendation includes what EAMD does and how to provide a set of security services for traffic using it.

Introduction

It is considered that the Internet of things (IoT) is one of the most important areas for future standardization. From the ITU-T perspective, IoT is defined as a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies in [b-ITU-T Y.2060].

One of the most relevant areas to IoT appears to be Ubiquitous Sensor Networks (USN). USN are networks of intelligent sensor nodes that could be deployed “anywhere, anytime, by anyone and anything”. We consider that security techniques for Ubiquitous Sensor Networks (USN) that have been investigated in Question 6 are effective in IoT because USN has many affinities with IoT is these techniques in the sense that it deals with devices such as sensing and actuating devices. With respect to USN security, Recommendations such as Security framework [ITU-T X.1311], Middleware security guidelines [ITU-T X.1312], Security requirements for wireless sensor network routing [ITU-T X.1313] and so on are already published. However, there has been no investigation of Recommendations on data confidentiality and integrity protection techniques that offer security for the device layer in USN. Therefore, the device layer security is a missing area in USN as well as in IoT hence this area should be studied and discussed for future standardization.

On the other hand, in certain IoT devices such as sensing and actuating devices which can be used in industrial control systems (ICS), there is a real-time processing requirement that tasks are processed within a certain period of time. To ensure data confidentiality and integrity protection, one of the most basic countermeasures is the application of data encryption/authentication algorithms. The problem with the standard applications of data encryption/authentication algorithms is that this requirement could not be met. The other problem is to integrate different security levels: more specifically, within a communication packet, data at different positions require different levels of importance consequently security. Therefore, encryption of data at the position that indicates a low security level is considered unnecessary processing overhead.

As mentioned above, to achieve the security for IoT devices, it is required a new application of data encryption/authentication algorithms that meets the real-time processing requirement and that integrate different security levels.

Therefore, the encryption with associated mask data that only encrypts data, within a communication packet, whose security level is high. The associated mask data is used to indicate the security levels of data at each position within a packet.

1 Scope

A scope of this Recommendation is to provide encryption procedure for Internet of things (IoT) device security. The procedure is intended to be applied to IoT devices that are devices with the mandatory capabilities of communication and the optional capabilities of sensing, actuation, data storage and data processing. This Recommendation provides specification of encryption with associated mask data (EAMD) for the IoT devices. It describes what EAMD does and how to provide a set of security services for traffic using it. Application examples are also provided in an Annex.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[IETF RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[IETF RFC4302] Kent, S., "IP Authentication Header", RFC 4302, December 2005.

[IETF RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.

The relevant ITU-T Recommendations are:

[ITU-T F.744] Service description and requirements for ubiquitous sensor network middleware

[ITU-T F.747.2] Deployment guidelines for ubiquitous sensor network applications and services for mitigating climate change

[ITU-T X.1311] Information technology - Security framework for ubiquitous sensor networks

[ITU-T X.1312] Ubiquitous sensor network middleware security guidelines

[ITU-T X.1313] Security requirements for wireless sensor network routing

[ITU-T X.usnsec-3] Secure routing mechanisms for WSN

[ITU-T Y.2002] Overview of ubiquitous networking and of its support in NGN

[ITU-T Y.2221] Requirements for support of ubiquitous sensor network (USN) applications and services in the NGN environment

[ITU-T Y.2060] Next Generation Networks – Frameworks and functional architecture models
Overview of the Internet of things

[ITU-T Y.2061] ITU-T Recommendation Y.2061 (2012), *Requirements for support of machine oriented communication applications in the NGN environment*

[ETSI TC ITS WG2] ITS WG2 - ETSI Portal

<http://portal.etsi.org/tb.aspx?tbid=708&SubTB=708>

[TSG GERAN] GSM/EDGE Radio Access Network

<http://www.3gpp.org/full-list/full-list/future/specifications-groups/tsg-geran>

<TBD>

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 Actuator [ITU-T Y.2061]: A device performing physical actions caused by an input signal.

NOTE - There are three types of actuators: **information actuators**, which are intended to provide visual, audio, sensory interaction with the human user; **gateway actuators**, which are intended to forward control commands given by SCN to other networks; **machine actuators**, which are electromechanical devices intended for physical interaction with the external environment.

3.1.2 Encapsulating Security Payload (ESP) [IETF RFC4303]: A member of the IPsec protocol suite that is used to provide confidentiality, data origin authentication, connectionless integrity, an anti-replay service (a form of fractional sequence integrity), and (limited) traffic flow confidentiality. The set of services provided depends on options selected at the time of Security Association establishment and on the location of the implementation in a network topology.

3.1.3 Security Parameters Index (SPI) [IETF RFC4301]: An arbitrary 32-bit value that is used by a receiver to identify the SA to which an incoming packet should be bound.

3.1.4 Sensed data [b-ITU-T F.744]: Data sensed by a sensor that is attached to a specific sensor node.

3.1.5 Sensor [b-ITU-T Y.2221]: An electronic device that senses a physical condition or chemical compound and delivers an electronic signal proportional to the observed characteristic.

3.1.6 Sequence Number [IETF RFC4303]: A counter value that increases by one for each packet sent, i.e., a per-SA packet sequence number.

<TBD>

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 controller : An electronic device to control actuators based on sensed data from sensors.

3.2.2 Security Association with Mask (SAM): The definition of a Security Association with Mask (SAM) is a security-protocol-specific set of parameters. SAM defines the services and mechanisms necessary to protect traffic by applying EAMD. The SAM is referred to by its associated protocol, depending on the protocol layers such as transport layer or IP layer. Algorithm identifiers, modes, layer identifier at which EAMD is applied, and cryptographic keys can be included in these parameters.

<TBD>

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AES	Advanced Encryption Standard
ESP	Encapsulating Security Payload
IP	Internet Protocol
IoT	Internet of things
IV	Initial Vector
MAC	Message Authentication Code
M2M	Machine to Machine
RFU	Reserved for Future Use
SA	Security Association with Mask
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
XOR	Exclusive OR

5 Conventions

None.

6 Introduction of encryption with associated mask data (EAMD)

6.1 What encryption with associated mask data does

The encryption with associated mask data performs cryptographic operations fractionally on the plain communication packet transmitted between some devices in the IoT devices. Cryptographic operations include encryption/decryption and MAC generation/verification. In the communication packet, data blocks on which cryptographic operations performed are indicated by a called *mask*.

It is supposed that sender knows algorithm of encryption, key, initial vector and mask for EAMD-secured communication and it is also supposed that sender and receiver know the identifier of algorithm and key each other. In this condition, the EAMD-secured communication is performed by getting the information for encryption with associated mask data at the sender and sharing the encryption information with receiver. Figure 1 illustrates the overview of encryption with associated mask data.

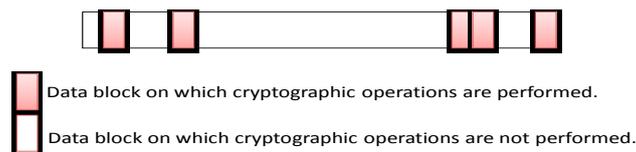


Fig. 1 The packet transmitted on communication using encryption with associated mask data

In an EAMD-secured communication, the outbound processing proceeds as follows:

1. Add a necessary padding for encryption.
2. Extract data for encryption using the mask for encryption and copy it into the buffer, which is used for temporary computations.
3. Encrypt the result in the buffer, using the key, encryption algorithm, any required data.
4. Substitute the result into the packet using the mask.
5. Encapsulate the result into the payload field.

If integrity is selected, the processing proceeds as follows:

6. Extract data for MAC generation using the mask for MAC generation and copy it into the buffer.
7. Add a necessary padding for MAC generation.
8. Generate the MAC over the result in the buffer.
9. Add the MAC to the packet.

Fig 2 illustrates the outbound processing.

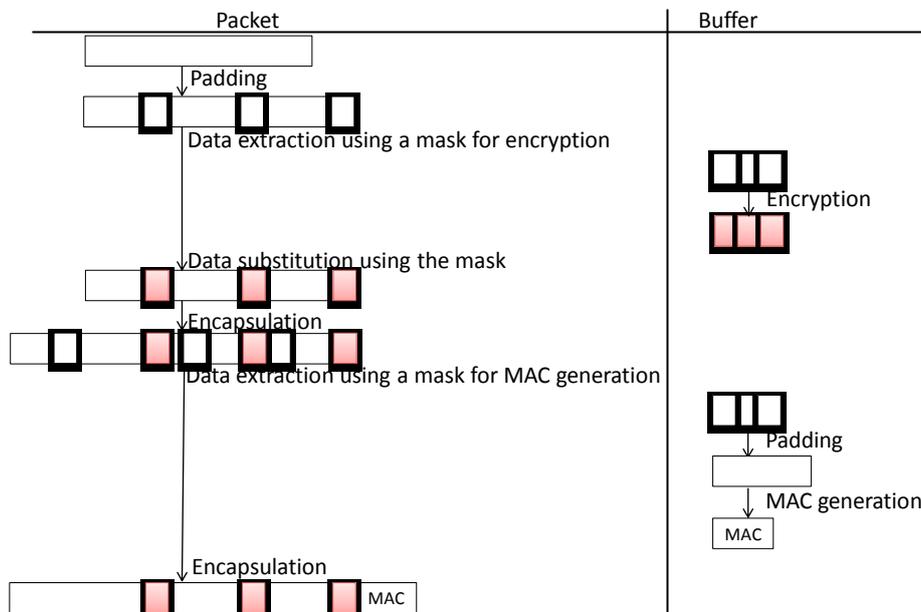


Fig. 2 How to generate a packet using encryption with associated mask data during outbound processing

In an EAMD-secured communication, the inbound processing proceeds as follows:

The inbound processing proceeds as follows:

If integrity is selected, the following steps 1 to 3 are performed:

1. Extract data from the packet minus the MAC into the buffer according to the mask for MAC generation
2. Add a necessary padding for MAC generation.
3. Compute the MAC over it using the specified integrity algorithm and verifies that it is the same as the MAC carried in the packet. If the computed and received MACs match, then the packet is valid, and it is accepted. If the test fails, then the receiver MUST discard the received packet as invalid.

4. Remove the header from the packet.
5. Extract data from the result into the buffer according to the mask for decryption.
6. Decrypt the extracted result in the buffer
7. Substitute the result in the buffer into the packet using the mask for decryption
8. Remove the padding for encryption from the packet

Fig 3 illustrates the inbound processing.

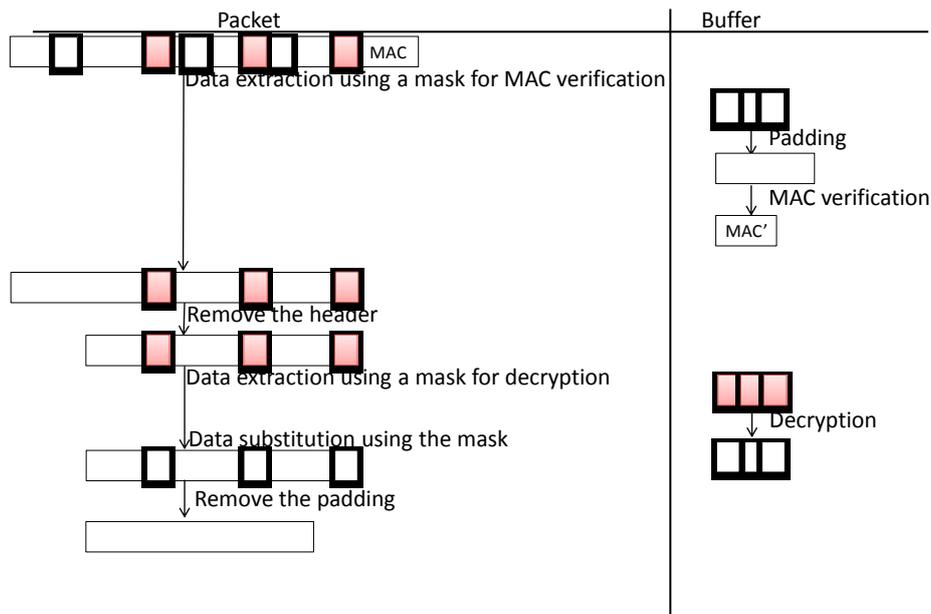


Fig. 3 How to generate a packet using encryption with associated mask data during inbound processing

6.2 Mask for Extracting target data for encryption with associated mask data

In operations for encryption with associated mask data, the target of block input to the corresponding algorithm is extracted by splitting the packet into the block size of using encryption algorithm according to the mask parameter.

7 How encryption with associated mask data does

This chapter describes how to provide a set of security services for traffic at each layer. This Recommendation describes a secure communication using encryption with associated mask data that is based on EAMDSP. The overview of this communication is described in Fig.4. The detail flow of the EAMD-secured communication is described in the following.

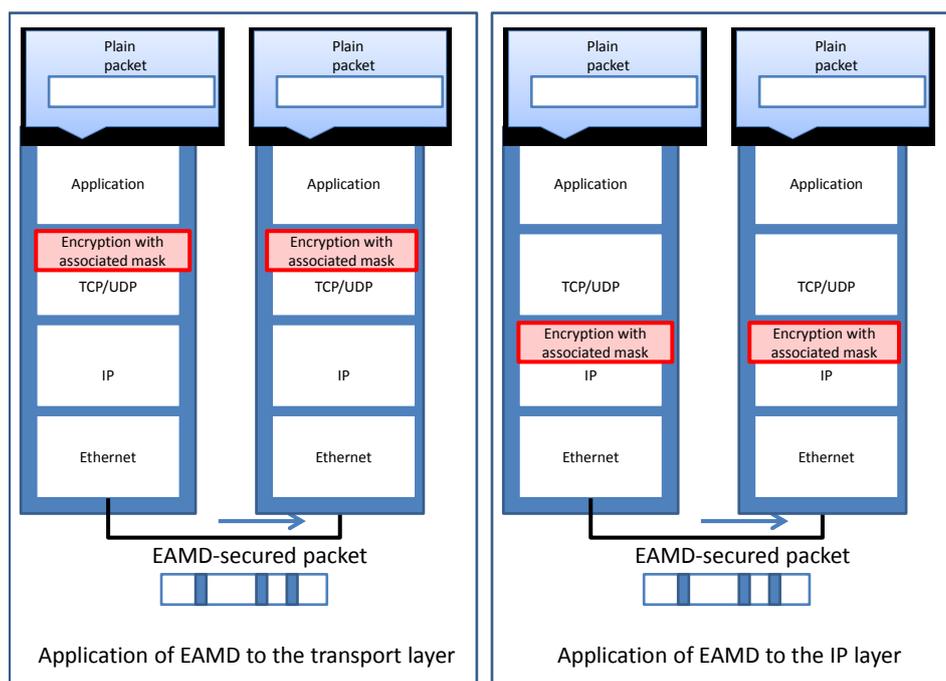


Fig. 4 Overview of communication using encryption with associated mask data (EAMD)

7.1 Security Association with Mask (SAM)

The definition of a Security Association with Mask (SAM) is a security-protocol-specific set of parameters. SAM defines the services and mechanisms necessary to protect traffic by applying EAMD. The SAM is referred to by its associated protocol, depending on the protocol layers such as transport layer or IP layer. Algorithm identifiers, modes, layer identifier at which EAMD is applied, layer-specific-parameter such as IP address and port, and cryptographic keys can be included in these parameters. State data associated with an SAM is represented in the SAM Database (SAMDB).

In this format, each mandatory parameter is described in Table 1.

Table 1 Mandatory parameters in CryptCtx in SA

No	Parameter	Meaning
1	encAlg	algorithm identifier for encryption
2	encKey	key for encryption
3	encMask	Area of being encrypted

Each optional parameter is described in Table 2

Table 2 Optional parameters in CryptCtx in SA

No	Parameter	Meaning
1	encRoundKey	round key for encryption
2	decRoundKey	round key for decryption
3	encIV	initial vector for encryption
4	macRoundKey	round key for MAC
5	macK1	Sub key for CMAC K1

6	macK2	Sub key for CMAC K2
7	KeyStream	Random numbers generated in advance
8	KeyStreamHead	Pointer to the head of unused random numbers
9	KeyStreamTail	Pointer to the tail of unused random numbers
10	EncIVTail	initial vector for random number generation
11	macAlg	algorithm identifier for MAC
12	macKey	key for MAC
13	macMask	Area of being used to generate MAC by designated algorithm

7.2 The Packet Format of EAMD Security Payload (EAMDSP)

Figure 5 illustrates a format of an EAMDSP (EAMD Security Payload) packet. The packet begins with the EAMDSP header of variable length. Following these fields is the Payload Data, which has substructure that depends on the choice of encryption algorithm and mode. Following the Payload Data are Padding and Pad Length fields, and the Next Header field. The optional Message authentication code (MAC) field completes the packet. The EAMDSP trailer consists of the Padding, Pad Length, and Next Header fields.

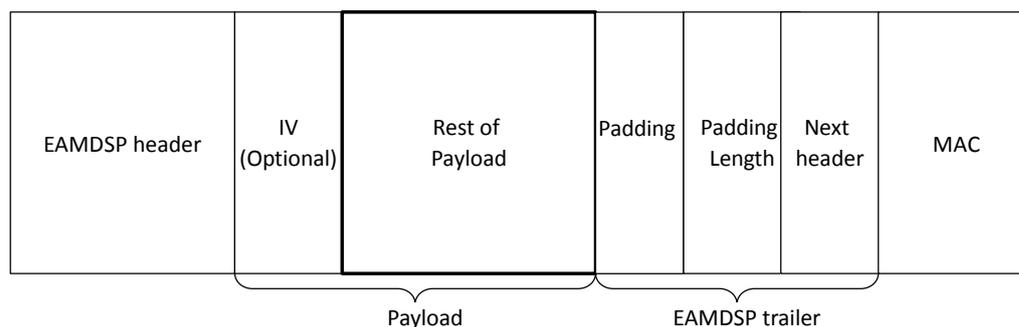


Fig. 5 Format of an EAMDSP Packet

The (transmitted) EAMDSP trailer consists of the Padding, Pad Length, and Next Header fields. Additional, implicit EAMDSP trailer data (which is not transmitted) is included in the integrity computation.

If the integrity service is selected, the integrity computation encompasses the EAMDSP header, Payload Data, and the EAMDSP trailer. If the confidentiality service is selected, the ciphertext consists of the Payload Data (except for any cryptographic synchronization data that may be included) and the EAMDSP trailer.

The following subchapters describe the fields in the header format. "Optional" means that the field is omitted if the option is not selected, i.e., it is present in neither the packet as transmitted nor as formatted for computation of an MAC. Whether or not an option is selected is determined as part of Security Association with Mask (SAM) establishment. Thus, the format of EAMDSP packets for a given SAM is fixed, for the duration of the SAM. In contrast, "mandatory" fields are always present in the EAMDSP packet format, for all SAMs.

7.2.1 Payload Data

Payload Data is a variable-length field containing data (from the original packet) described by the Next Header field. The Payload Data field is mandatory and is an integral number of bytes in length. If the algorithm used to encrypt the payload requires cryptographic synchronization data, e.g., an Initialization Vector (IV), then this data is carried explicitly in the Payload field, but it is not called out as a separate field in EAMDSP, i.e., the transmission of an explicit IV is invisible to EAMDSP.

7.2.2 Padding (for Encryption)

If an encryption algorithm is employed that requires the plaintext to be a multiple of some number of bytes, e.g., the block size of a block cipher, the Padding field is used to fill the plaintext (consisting of the Payload Data, Padding, Pad Length, and Next Header fields) to the size required by the algorithm.

7.2.3 Pad Length

The Pad Length field indicates the number of pad bytes immediately preceding it in the Padding field. The Pad Length field is mandatory.

7.2.4 Next Header

The Next Header is a mandatory. This field identifies the type of data contained in the Payload Data field, e.g., a next layer header and data.

7.2.5 Message authentication code (MAC)

The Message authentication code is a variable-length field computed over the data that the mask indicates to protect in terms of integrity. Implicit EAMDSP trailer fields such as padding for MAC generation are included in the MAC computation. The MAC field is optional. It is present only if the integrity service is selected and is provided by either a separate integrity algorithm or a combined mode algorithm that uses an MAC. The length of the field is specified by the integrity algorithm selected and associated with the SAM. The integrity algorithm specification **MUST** specify the length of the MAC and the comparison rules and processing steps for validation.

7.3 Packet Processing

7.3.1 Outbound Packet Processing

The outbound processing using encryption with associated mask data proceeds as follows.

1) SAM Lookup

Before EAMDSP is applied to an outbound packet, the associated SAM that calls for EAMDSP processing is determined according to the information such as the layer identifier and the layer-specific parameter such as IP address or port number in the packet. SAM indicates key and masks for encryption and for MAC generation.

2) Data transformation using EAMD described in 6.1

3) Packet Sending

The original header is added to the EAMD-transformed packet and send the resulting packet out to the network.

7.3.2 Inbound Packet Processing

The inbound processing using encryption with associated mask data proceeds as follows.

1) SAM Lookup

Upon receipt of a packet containing an EAMDSP Header, the receiver determines the appropriate SAM via lookup in the SAMD. The SAMD entry for the SAM also indicates which layer EAMD was applied during outbound processing and the layer-specific parameter such as IP address or port number in the packet, and whether the MAC field should be present. Also, the SAMD entry will specify the algorithms and keys to be employed for decryption and MAC verification (if applicable).

2) EAMDSP header data verification.

EAMDSP header data check can be effected by using the certain values in the EAMDSP Header and is performed prior to integrity checking and decryption. If this check fails, the packet is discarded.

3) Data transformation using EAMD described in 6.1

Annex 1

Bindings to existing protocols

To use a secure communication using encryption with associated mask data in practice, the layer where it is applied must be fixed. There are several options for this, such as the transport layer, and the IP layer, and so on. This annex describes how to bind encryption with associated mask data to existing protocols.

A3.1 Binding to UDP protocol [IETF RFC768]

A3.1.1 SAM format

SAM defines the services and mechanisms necessary to protect traffic by applying EAMD. If EAMD is applied to the transport layer, a Security Association with Mask (SAM) format is described in Fig. 6.

```
CryptCtx ::= SEQUENCE {
    encAlg OCTET STRING (SIZE (4))
    encKey OCTET STRING (SIZE (KEY_SIZE_MAX))
    encMask OCTET STRING (SIZE (MASKLEN))
}
SecurityAssertion ::= SEQUENCE {
    LayerIdentifier OCTET STRING(SIZE(1))
    IPAddr OCTET STRING (SIZE (4))
    Port OCTET STRING (SIZE (4))
    CryptCtx CryptCtx
}
```

Fig. 6 SAM Format for binding to UDP

A3.1.2 Packet format

The packet begins with the EAMDSP header of variable length. Following these fields is the Payload Data, which has substructure that depends on the choice of encryption algorithm and mode. Following the Payload Data are Padding and Pad Length fields, and the Next Header field. The optional Message authentication code (MAC) field completes the packet. The EAMDSP trailer consists of the Padding, Pad Length, and Next Header fields.

A3.1.3 Packet Processing

The outbound processing using encryption with associated mask data proceeds as follows.

1. SAM Lookup
the associated SAM that calls for EAMDSP processing is determined according to the information such as the layer identifier and the IP address and port number in the packet.
2. Data transformation using EAMD
Encryption and MAC generation are performed using EAMD according to the process in 6.1.
3. Packet Sending
The original header is added to the EAMD-transformed packet and send the resulting packet out to the network.

The inbound processing using encryption with associated mask data proceeds as follows.

1. SAM Lookup
the associated SAM that calls for EAMDSP processing is determined according to the information such as the layer identifier that identifies the transport layer and IP address and the port number in the packet.
2. Data transformation using EAMD
MAC verification and decryption are performed using EAMD according to the process in 6.1.

<TBD>

A3.2 Binding to the IPsec ESP Protocol [IETF RFC4303]

A3.2.1 SAM format

SAM defines the services and mechanisms necessary to protect traffic by applying EAMD. If EAMD is applied to the transport layer, a Security Association with Mask (SAM) format is described in Fig. 7.

```
CryptCtx ::= SEQUENCE {  
    encAlg OCTET STRING (SIZE (4))  
    encKey OCTET STRING (SIZE (KEY_SIZE_MAX))  
    encMask OCTET STRING (SIZE (MASKLEN))  
}  
SecurityAssertion ::= SEQUENCE {  
    LayerIdentifier OCTET STRING(SIZE(1))  
    SPI OCTET STRING (SIZE (4))
```



Fig. 7 SAM Format for the IP layer

A3.2.2 Packet format

Figure 8 illustrates a format example of an EAMDSP (EAMD Security Payload) packet. The packet begins with the EAMDSP header of variable length. Following these fields is the Payload Data, which has substructure that depends on the choice of encryption algorithm and mode. Following the Payload Data are Padding and Pad Length fields, and the Next Header field. The optional Message authentication code (MAC) field completes the packet. The EAMDSP trailer consists of the Padding, Pad Length, and Next Header fields. Considering the alignment due to EAMD MAC computations and EAMD-encryption, in another format example, Sequence No. length can be 8B and the next Header field placed in the EAMDSP header.

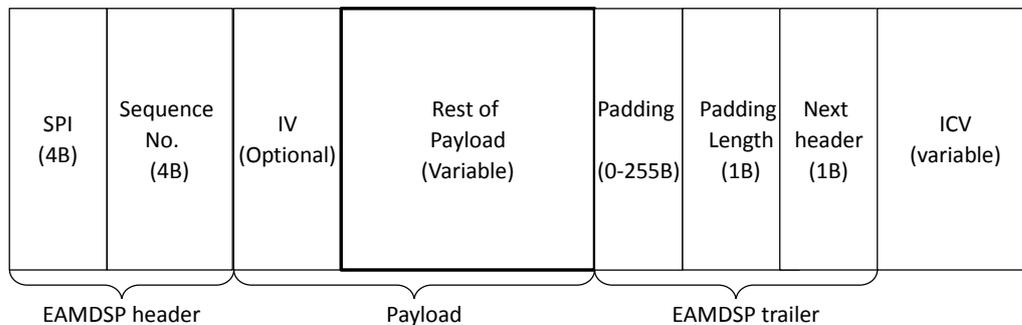


Fig. 8 Format example of an EAMDSP Packet for binding to IPsec ESP protocol

1) Security Parameters Index (SPI)

The SPI is an arbitrary 32-bit value that is used by a receiver to identify the SAM to which an incoming packet is bound. The SPI field is mandatory. The SPI is carried in the protocol to enable the receiving system to select the SAM under which a received packet will be processed.

2) Sequence Number

This unsigned 32-bit field or 64-bit field contains a counter value that increases by one for each packet sent, i.e., a per-SAM packet sequence number or, alternatively, a value which is generated according to an unambiguous rule.

3) Payload Data

Payload Data is a variable-length field containing data (from the original packet) described by the Next Header field. The Payload Data field is mandatory and is an integral number of bytes in length. If the algorithm used to encrypt the payload requires cryptographic synchronization data, e.g., an Initialization Vector (IV), then this data is carried explicitly in the Payload field, but it is not called out as a separate field in EAMDSP, i.e., the transmission of an explicit IV is invisible to EAMDSP.

4) Padding (for Encryption)

Padding also may be required, irrespective of encryption algorithm requirements, to ensure that the resulting ciphertext terminates on a 4-byte boundary. Specifically, the Pad Length and Next Header fields must be right aligned within a 4-byte word, as illustrated in the EAMDSP packet format figures above, to ensure that the MAC field (if present) is aligned on a 4-byte boundary.

5) Pad Length

The Pad Length field indicates the number of pad bytes immediately preceding it in the Padding field. The range of valid values is 0 to 255, where a value of zero indicates that no Padding bytes are present. The Pad Length field is mandatory.

6) Next Header

The Next Header is a mandatory, 8-bit field that identifies the type of data contained in the Payload Data field, e.g., a next layer header and data.

7) Message authentication code (MAC)

The Message authentication code is a variable-length field computed over the data that the mask indicates to protect in terms of integrity. Implicit EAMDSP trailer fields such as padding for MAC generation are included in the MAC computation. The MAC field is optional.

A3.2.3 Packet Processing

The outbound processing using encryption with associated mask data proceeds as follows.

1) SAM Lookup

the associated SAM that calls for EAMDSP processing is determined according to the information such as the layer identifier and the layer-specific parameter such as IP address or port number in the packet.

2) Data transformation using EAMD

Encryption and MAC generation are performed using EAMD according to the process in 6.1.

3) Packet Sending

The original header is added to the EAMD-transformed packet and send the resulting packet out to the network.

The inbound processing using encryption with associated mask data proceeds as follows.

1) SAM Lookup

the associated SAM that calls for EAMDSP processing is determined according to the information such as the layer identifier that identifies the transport layer and IP address and the port number in the packet.

2) Sequence Number Verification

The Sequence Number check is effected by using the Sequence Number value in the EAMDSP Header and is performed prior to integrity checking and decryption. If this check fails, the packet is discarded.

3) Data transformation using EAMD

MAC verification and decryption are performed using EAMD according to the process in 6.1.

<TBD>

A3.3 Mask for Extracting target data for encryption with associated mask data

In operations for encryption with associated mask data, the target of block input to the corresponding algorithm is extracted by splitting the packet into the block size of using encryption algorithm according to the mask parameter. For example, in the case of encryption with associated mask data using AES, the payload is split into every 128 bits because the block length of AES is 128 bits. And the target of decryption block is extracted by finding the block according to the mask. After that, the target data for the operation is generated by concatenating the target of decryption block. The format of mask is described in Fig.9 and Fig.10. This parameter shows which block should be encrypted or decrypted in case of splitting the payload into the block size of using encryption algorithm.



Fig. 9 Format of mask

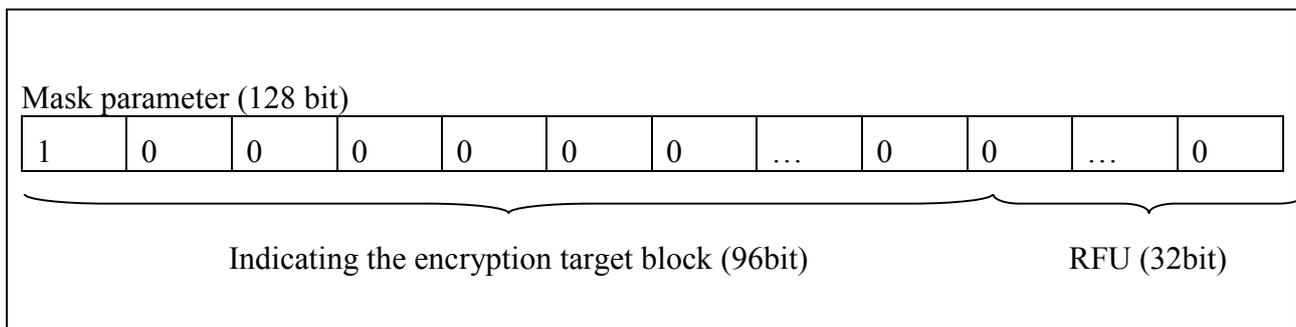


Fig. 10 Detail format of mask parameter

In this case, this mask parameter means that only first block is encrypted because the first bit of mask parameter is true. If you want to encrypt some areas, you should change the some bits of mask parameter from false to true.

A3.4 Padding Algorithm

A padding algorithm may be described as follows.

- Append an '0x80' at the end of payload
- If the length of payload is a multiple of block length of encryption algorithm, the padding is finished.

If the length of payload is NOT a multiple of block length of encryption algorithm, append an '0x00' at the end of payload until the length of payload is a multiple of it.

Bibliography

[IETF RFC2460] Deering, S., and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[IETF RFC791] Postel, J., "*Internet Protocol*", STD 5, RFC 791, September 1981.

[b-ITU-T Y.2221] ITU-T Recommendation Y.2221 (2010), *Requirements for support of ubiquitous sensor network (USN) applications and services in the NGN environment*

<TBD>
