International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.1362

(03/2017)

SERIES X: DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

Secure applications and services – Internet of things (IoT) security

Simple encryption procedure for Internet of things (IoT) environments

Recommendation ITU-T X.1362

International Telecommunication Union

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

PUBLIC DATA NETWORKS	X.1–X.199
OPEN SYSTEMS INTERCONNECTION	X.200–X.299
INTERWORKING BETWEEN NETWORKS	X.300–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	X.600–X.699
OSI MANAGEMENT	X.700–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	X.850–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
INFORMATION AND NETWORK SECURITY	
General security aspects	X.1000–X.1029
Network security	X.1030–X.1049
Security management	X.1050–X.1069
Telebiometrics	X.1080–X.1099
SECURE APPLICATIONS AND SERVICES	
Multicast security	X.1100–X.1109
Home network security	X.1110–X.1119
Mobile security	X.1120–X.1139
Web security	X.1140–X.1149
Security protocols	X.1150–X.1159
Peer-to-peer security	X.1160–X.1169
Networked ID security	X.1170–X.1179
IPTV security	X.1180–X.1199
CYBERSPACE SECURITY	
Cybersecurity	X.1200–X.1229
Countering spam	X.1230–X.1249
Identity management	X.1250–X.1279
SECURE APPLICATIONS AND SERVICES	
Emergency communications	X.1300–X.1309
Ubiquitous sensor network security	X.1310–X.1339
PKI related Recommendations	X.1340–X.1349
Internet of things (IoT) security	**X.1360–X.1369**
Intelligent transportation system (ITS) security	X.1370–X.1379
CYBERSECURITY INFORMATION EXCHANGE	
Overview of cybersecurity	X.1500–X.1519
Vulnerability/state exchange	X.1520–X.1539
Event/incident/heuristics exchange	X.1540–X.1549
Exchange of policies	X.1550–X.1559
Heuristics and information request	X.1560–X.1569
Identification and discovery	X.1570–X.1579
Assured exchange	X.1580–X.1589
CLOUD COMPUTING SECURITY	
Overview of cloud computing security	X.1600–X.1601
Cloud computing security design	X.1602–X.1639
Cloud computing security best practices and guidelines	X.1640–X.1659
Cloud computing security implementation	X.1660–X.1679
Other cloud computing security	X.1680–X.1699

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T X.1362

Simple encryption procedure for Internet of things (IoT) environments

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 environments, especially for IoT devices, there is a real-time processing requirement where 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.

Recommendation ITU-T X.1362 specifies encryption with associated mask data (EAMD) for the Internet of things devices. It describes EAMD and how it provides a set of security services for traffic using EADM.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T X.1362	2017-03-30	17	11.1002/1000/13196

Keywords

Application of data encryption/authentication algorithms, encryption with associated mask data EAMD, IoT devices, IoT environments, real-time processing requirement.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

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].

Ubiquitous sensor networks (USNs) appears to be one of the most relevant areas of IoT. USNs 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 are effective in IoT because USN has many affinities with IoT in the sense that it deals with devices such as sensing and actuating devices. With respect to USN security, Recommendations such as security framework [b-ITU-T X.1311], middleware security guidelines [b-ITU-T X.1312], and security requirements for wireless sensor network routing [b-ITU-T X.1313] 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 environments, especially for IoT devices such as sensing and actuating devices which can be used in industrial control systems (ICSs), there is a real-time processing requirement where 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 important consequential security. Therefore, encryption of data at the position that indicates a low security level is considered as an unnecessary processing overhead.

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

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

Recommendation ITU-T X.1362

Simple encryption procedure for Internet of things (IoT) environments

1 Scope

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

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 RFC 4303] IETF RFC 4303 (2005), *IP Encapsulating Security Payload (ESP).*

[IETF RFC 7296] IETF RFC 7296 (2014), *Internet Key Exchange Protocol Version 2 (IKEv2).*

[IETF RFC 7321] IETF RFC 7321 (2014), *Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH).*

[ISO/IEC 10116] ISO/IEC 10116:2006, *Information technology – Security techniques –Modes of operation for an n-bit block cipher.*

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 actuator [b-ITU-T Y.4109]: A device performing physical actions caused by an input signal.

NOTE – As examples, an actuator might act on the flow of a gas or liquid, on electricity distribution, or through a mechanical operation. Dimmers and relays are examples of actuators. The decision to activate the actuator may come from an MOC application, a human or MOC devices and gateways.

3.1.2 encapsulating security payload (ESP) [IETF RFC 4303]: An IPsec protocol 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 (SA) establishment and on the location of the implementation in a network topology.

3.1.3 security parameters index (SPI) [b-IETF RFC 4301]: 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.4104]: Data sensed by a sensor that is attached to a specific sensor node.

3.1.5 sensor [b-ITU-T Y.4105]: 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 RFC 4303]: This unsigned 32-bit field contains a counter value that increases by one for each packet sent, i.e., a per-SA packet sequence number.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

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

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

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
CMAC	Cipher-based Message Authentication Code
EAMD	Encryption with Associated Mask Data
EAMDSP	EAMD Security Payload
ESP	Encapsulating Security Payload
ICS	Industrial Control System
IP	Internet Protocol
IPSec	IP Security
IoT	Internet of Things
IV	Initialization Vector
MAC	Message Authentication Code
SA	Security Association
SAM	Security Association with Mask
SAMD	SAM Database
SPI	Security Parameters Index
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
USN	Ubiquitous Sensor Networks
XOR	Exclusive OR

5 Conventions

None.

6 Introduction of encryption with associated mask data (EAMD)

6.1 Specification of the EAMD procedure

There is a wide range of security threats on IoT environments [b-ZT]. This Recommendation focuses on the following threats:

1) Impersonation attacks that intercept legitimate data or fake legitimate data that lead to information disclosure or tampering.
2) Eavesdropping attacks that capture packets from the network transmitted by the computers of others, and reading the data content in search of any kind of confidential information.
3) Spoofing attacks that disguise as a legitimate component to obtain data or information tampering.

To mitigate these threats on IoT environments, especially for IoT devices, this Recommendation specifies the encryption with associated mask data (EAMD) that performs cryptographic operations fractionally on plain communication packets transmitted between some devices. Cryptographic operations include encryption/decryption and message authentication code (MAC) generation/verification, and authenticated encryption. The cryptographic algorithms employed in EAMD are out of the scope of this Recommendation. However [b-ISO/IEC 9797], [b-ISO/IEC 18033] and [b-ISO/IEC 19772] are good references for encryption algorithms, MAC algorithms and authenticated encryption algorithms, respectively.

Note that the construction described in this clause can be viewed as a protocol using encrypt-then-MAC [b-ASIACRYPT], and [b-EUROCRYPT].

In the communication packet, data blocks on which cryptographic operations are performed are indicated by a mask.

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

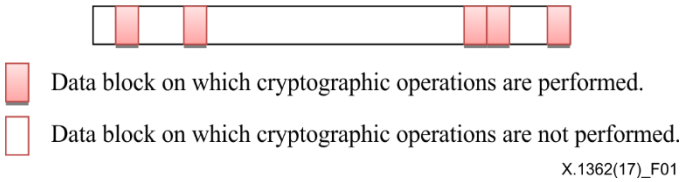

Figure 1 – Packet transmitted on communication using encryption with associated mask data

In an EAMD-secured communication, the outbound processing is 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, and 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, processing is as follows:[1]

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 MAC over the result in the buffer.

9) Add MAC to the packet.

Figure 2 illustrates the outbound processing.

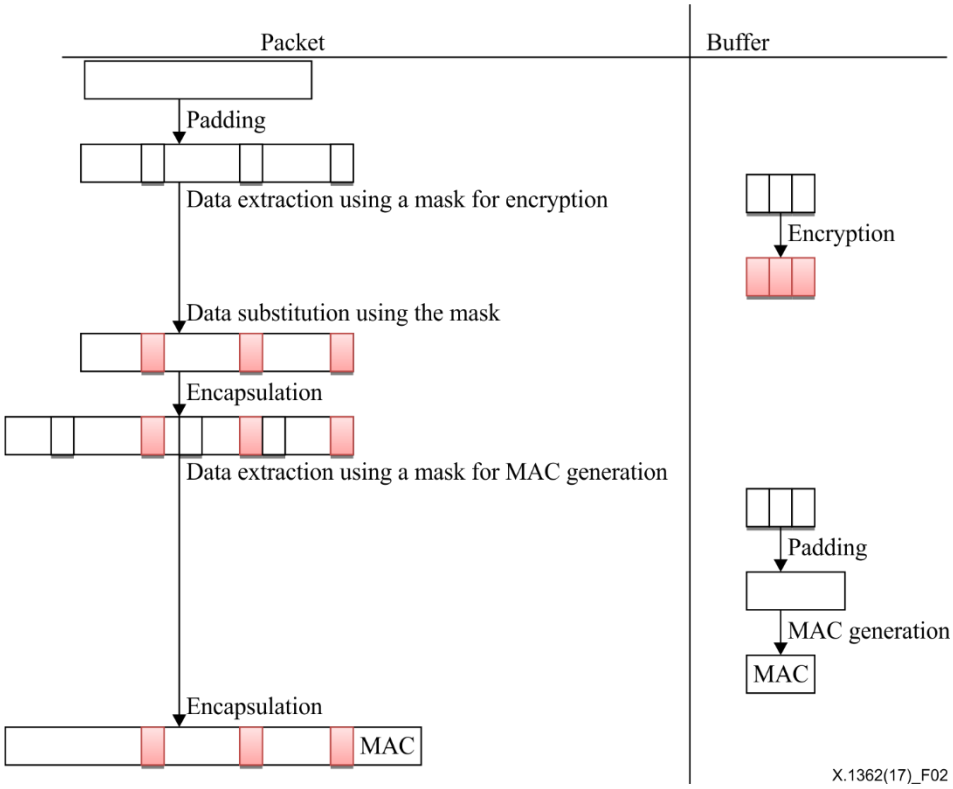

Figure 2 – Generating a packet using encryption with associated mask data during outbound processing

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

If integrity is selected, steps 1 to 3 below are performed[1]:

1) Extract data from the packet minus MAC into the buffer according to the mask for MAC generation.

2) Add a necessary padding for MAC generation.

3) Compute MAC over the padded data using the specified integrity algorithm and verify 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 shall 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.

---

[1] To use encryption with associated mask data binding with IPsec protocol, confidentiality with authentication should be ensured.

7) Substitute the result in the buffer into the packet using the mask for decryption.

9) Remove the padding for encryption from the packet.

Figure 3 illustrates the inbound processing.

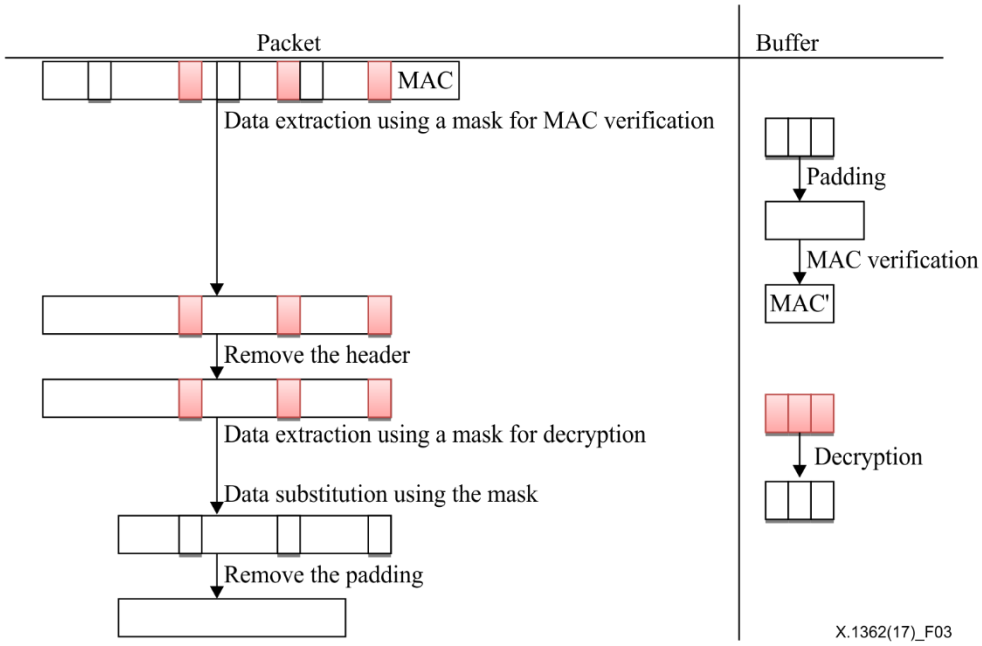

Figure 3 – Generating a packet using encryption with associated mask data during inbound processing

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 that uses encryption algorithm according to the mask parameter.

7 Encrypt with associated mask data

This clause 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 EAMD security payload (EAMDSP). The overview of this communication is described in Figure 4. The detailed flow of the EAMD-secured communication is described as follows:

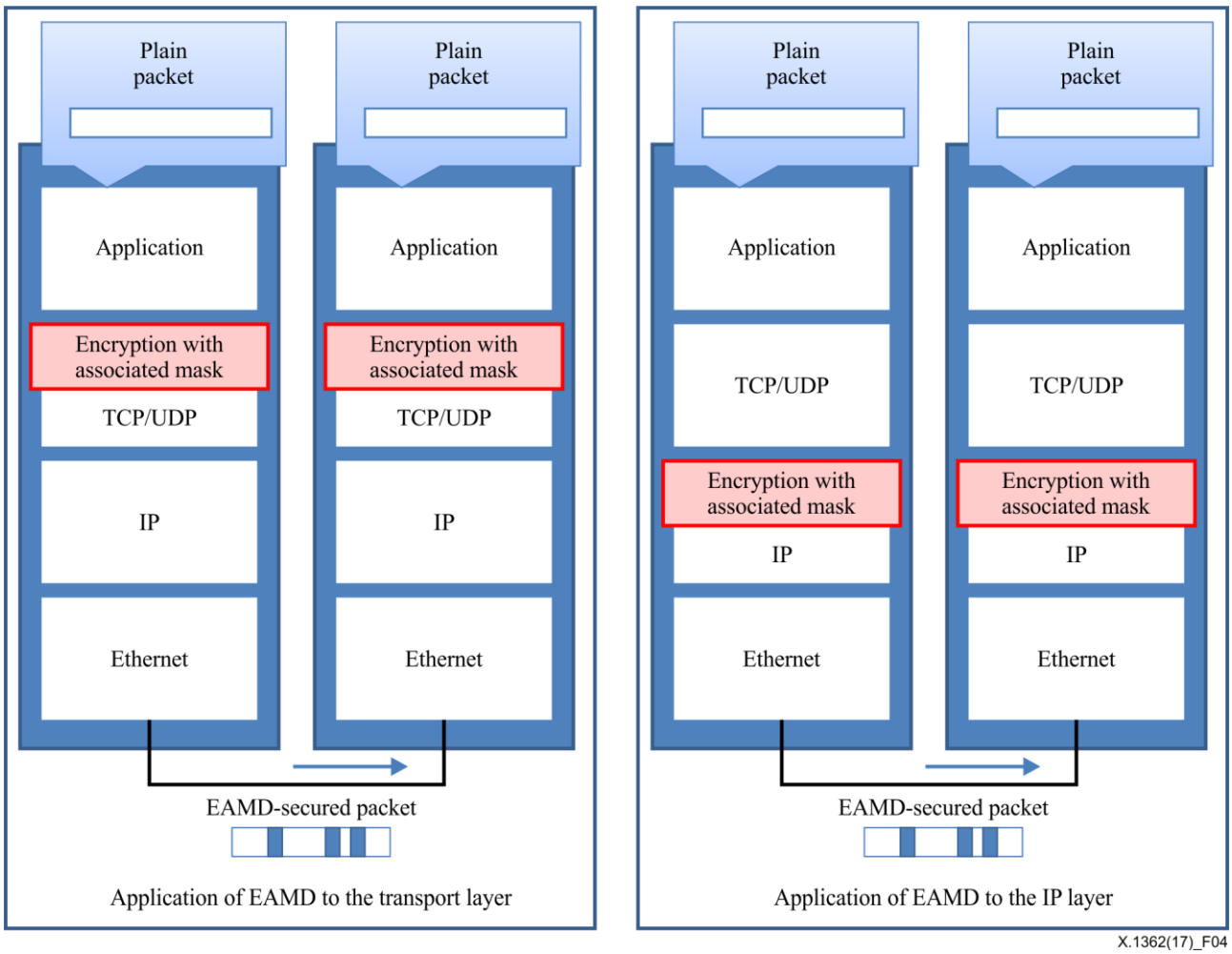

Figure 4 – Overview of communication using encryption with associated mask data (EAMD)

7.1 Security association with mask (SAM)

Security association with mask (SAM) is defined as a security-protocol-specific set of parameters. SAM defines the services and mechanisms necessary to protect traffic by applying EAMD. SAM is referred to by its associated protocol, depending on the protocol layers such as transport layer or Internet protocol (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. SAM contains CryptCtx defined as a set of cryptographic parameters. State data associated with SAM is represented in the SAM database (SAMD).

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

Table 1 – Mandatory parameters in CryptCtx in security association (SA)

No.	Parameter	Meaning
1	encAlg	Algorithm identifier for encryption
2	encKey	Key for encryption
3	encMask	Area that is 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 (IV) 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 that is being used to generate MAC by a designated algorithm

7.2 Packet format of EAMD security payload (EAMDSP)

Figure 5 illustrates a format of an EAMD security payload (EAMDSP) packet. The packet begins with the EAMDSP header of variable length. Following this field is the payload data, which has a substructure that depends on the choice of encryption algorithm and mode. Following the payload data are the 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.

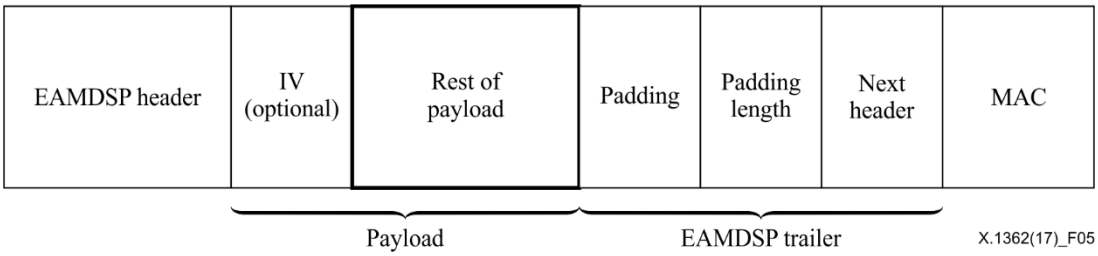

Figure 5 – Format of an EAMDSP packet

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 clauses 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 the computation of MAC. Whether or not an option is selected is determined as part of the SAM establishment. Thus, the format of EAMDSP packets for a given SAM is fixed for the duration of 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 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 MAC. The length of the field is specified by the integrity algorithm selected and associated with SAM. The integrity algorithm specification shall specify the length of 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 is 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 some information like the layer identifier and the layer-specific parameter such as IP address or port number in the packet. SAM indicates the key and masks for encryption and for MAC generation.

2) Data transformation using EAMD is described in clause 6.1.

3) Packet sending:

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

7.3.2 Inbound packet processing

The inbound processing using encryption with associated mask data is as follows:

1) SAM lookup:

   Upon receipt of a packet containing an EAMDSP header, the receiver determines the appropriate SAM via lookup in SAMD. The SAMD entry for SAM also indicates which layer EAMD is applied during the 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. In addition, 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 certain values in the EAMDSP header and is performed prior to integrity checking and decryption. If this check fails, the packet is discarded.

Data transformation using EAMD is described in clause 6.1.

8 EAMD employing an authenticated encryption algorithm

8.1 Security association with mask (SAM)

In the case of EAMD employing an authenticated encryption algorithm, SAM is also defined as in clause 7.1.

In this format, each mandatory parameter is described in Table 3.

Table 3 – Mandatory parameters in CryptCtx in SA

No.	Parameter	Meaning
1	auencAlg	Algorithm identifier for authenticated encryption
2	auencKey	Key for authenticated encryption
3	encMask	Area that is being encrypted

Each optional parameter is described in Table 4.

Table 4 – Optional parameters in CryptCtx in SA

No.	Parameter	Meaning
1	auencRoundKey	Round key for authenticated encryption
2	audecRoundKey	Round key for decryption
3	IV	Initial vector for authenticated encryption
4	Nonce	Round key for authenticated encryption

8.2 Packet format of EAMD security payload (EAMDSP)

Figure 6 illustrates a format of an EAMDSP (EAMD security payload) packet. The packet begins with the EAMDSP header of variable length. Following this field is the payload data, which has a substructure that depends on the choice of encryption algorithm and mode. Following the payload data are the padding and pad length fields, and the next header field. The EAMDSP trailer consists of the padding, pad length, and next header fields.

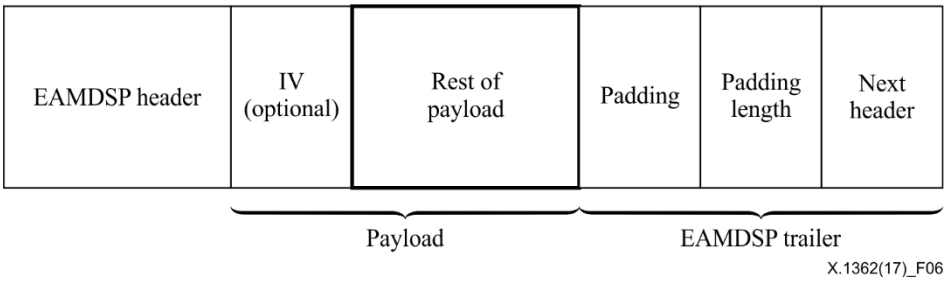

Figure 6 – Format of an EAMDSP (for authenticated encryption) packet without MAC

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 clauses 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 MAC. Whether or not an option is selected is determined as part of the SAM establishment. Thus, the format of EAMDSP packets for a given SAM is fixed for the duration of SAM. In contrast, "mandatory" fields are always present in the EAMDSP packet format for all SAMs.

8.2.1 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.

The format of the encapsulating security payload (ESP) packet can be expressed as ESP = SPI || Sequence Number || IV || C, where C is the ciphertext that the authenticated encryption algorithm produces. In this case, C incorporates the authentication tag.

8.2.2 Padding (for authenticated encryption)

If an authenticated 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.

8.2.3 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.

8.2.4 Next header

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

8.3 Packet processing

8.3.1 Outbound packet processing

The outbound processing using encryption with associated mask data is 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 some information like the layer identifier and the layer-specific parameter such as Internet protocol (IP) address or port number in the packet. SAM indicates the key and masks for authenticated encryption.

2) Data transformation using EAMD authenticated encryption mode

   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, and any required data.
   4) Substitute the ciphered text into the packet using the mask.
   5) Add the authentication tag to the packet as the MAC.

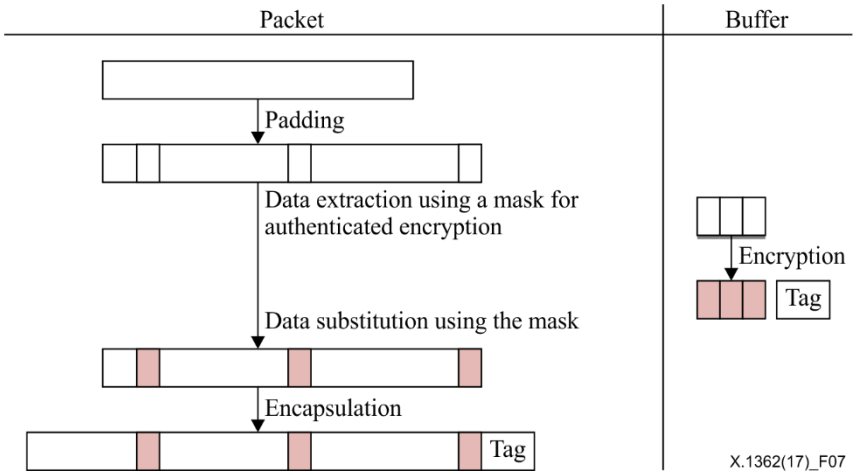

Figure 7 – Outbound packet processing for the authenticated encryption mode

3) Packet sending:

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

8.3.2 Inbound packet processing

The inbound processing using encryption with associated mask data is as follows:

1) SAM lookup:

   Upon receipt of a packet containing an EAMDSP header, the receiver determines the appropriate SAM via lookup in SAMD. The SAMD entry for SAM also indicates which layer EAMD had applied during the outbound processing and the layer-specific parameter such as IP address or port number in the packet. In addition, the SAMD entry will specify the algorithms and keys to be employed for decryption and tag verification.

2) EAMDSP header data verification:

   EAMDSP header data check can be effected by using 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 authenticated decryption mode

   1) Remove the header from the packet.
   2) Extract data for decryption using the mask for decryption, and copy it into the buffer, which is used for temporary computations.

3) Separate the authentication tag from the packet and copy it into the buffer.

4) Decrypt the result in the buffer, using the key, decryption algorithm, and any required data.

5) Substitute the result of the decryption into the packet using the mask if the decryption is not failed.

4) Remove the padding for encryption from the packet.

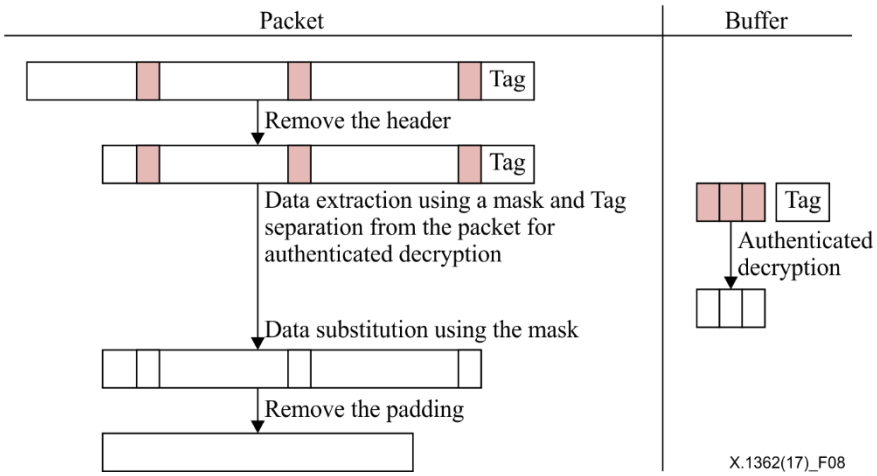

Figure 8 – Inbound packet processing for the authenticated encryption mode

9 Guidance and limitation

9.1 Guidance on SAM establishment

Concerning the EAMD encryption masking method, it should be noted that if a malicious entity were able to modify the mask, they could set the mask value such that none of the data gets encrypted. Once the mask is altered in this way, all of the device's data would be communicated "in the clear" (i.e., unencrypted). This would represent a significant security vulnerability.

To solve this problem, the following issues should be addressed:

1) Security of Mask communications

In order to initialize and update the keying material such as cryptographic keys, initial vectors, and other security parameters, there are a number of key establishment protocols such as Internet Key Exchange Protocol Version 2 (IKEv2) [IETF RFC 7296], Key agreement, and Key transport.

It should be ensured that the mask is initialized and updated in connection with the initialization and the update of the keying material during communication between related entities by using these protocols. For instance, during the key establishment communication process, the above keying material should contain the mask as well so that the integrity and confidentially of the mask can be achieved by means of encryption algorithms and MAC algorithms used in these protocols.

2) Security of Mask storages

It should be ensured that, once the mask is on the device, there is no protocol by which the other device can read the mask.

The following methods can be suggested for this purpose.

The first one is a secure system design that assigns the system components in such a way that EAMD-applied devices are not directly communicated to an entity outside of the

system, although a secure gateway component with a high computing capability is communicated to such entity.

The second method is a device protection by means of tamper-resistant hardware or the software obfuscation method that creates an obfuscated code that is difficult for humans to understand.

9.2 Guidance of the proper usage of initialization vectors and nonces

This clause provides guidance on the proper usage of initialization vectors (and additionally block cipher modes and padding). Improper usage of initialization vectors or padding is a common pitfall for attacks on protocols. An IV or a nonce will most likely play a crucial role in the security of the protocol.

In order to use the cipher block chaining (CBC) mode [ISO/IEC 10116] for encryption that features the combining of the plaintext blocks with the previous ciphertext blocks, the following should be done:

When CBC mode is used as the mode of operation of block ciphers, the security against the padding oracle attacks in [b-CBCPADD] should be considered. The CBC mode requires an IV to combine with the first plaintext block. IV needs not be secret, but it shall be unpredictable.

In order to use an authenticated encryption algorithm securely, the following should be done:

If an application cannot meet the uniqueness requirement on nonce generation, then it shall use a zero-length nonce. Randomized or stateful algorithms [b-IETF RFC 5116] are suitable for use with such applications. Otherwise, an application should use nonces with a length of twelve octets.

When nonces or IVs are repeated, many schemes have practical attacks which would reveal, for instance, the exclusive or (XOR) of two packets. Therefore, it is highly recommended that IV or nonce be guaranteed unique.

9.3 Limitation of the use of EAMD

The use of EAMD is limited in the system real-time performance requirements.

The merit of EAMD is optimized to the system where the senders and the receiver have a certain level of computing capability, e.g., the CPU architectures are 16-bit or 32-bit with a reasonable frequency (hundreds MHz) and memory.

It should be noted that EAMD might not be a good solution for the systems with power constrained requirements since power consumption for EAMD buffering process may result in a significant overhead.

It should be noted that the mask is very sensitive like cryptographic key as stated in the above, hence EAMD can be applied only when there is an assumption that the mask is securely managed and protected.

Annex A

Bindings to existing protocols

(This annex forms an integral part of this Recommendation.)

For a secure communication using encryption with associated mask data, the layer where encryption is applied shall be fixed. There are several possible layers where this can be done, such as the transport layer, the IP layer, etc. This annex describes how to bind encryption with associated mask data to existing protocols. The use of encryption with associated mask data binding with IPsec protocol needs to provide confidentiality and authentication. Confidentiality with authentication should be ensured [IETF RFC 7321].

A.1 Binding to the IP security (IPSec) ESP protocol IETF RFC 4303

A.1.1 SAM format

SAM defines the services and mechanisms necessary to protect traffic by applying EAMD. If EAMD is applied to the network layer, a security association with mask (SAM) format is described in Figure A.1.

```
SecurityAssertion ::= SEQUENCE {
  layerIdentifier OCTET STRING (SIZE(1)),
  SPI             OCTET STRING (SIZE (4)),
  ipAddr          OCTET STRING (SIZE (4)),
  cryptCtx        CryptCtx
}
CryptCtx ::= SEQUENCE {
  encAlg    OCTET STRING (SIZE (4))
  encKey    OCTET STRING (SIZE (keySizeMax)),
  encMask   OCTET STRING (SIZE (maskLength))
}
keySizeMax INTEGER ::= 64
maskLength INTEGER ::= 16
```

Figure A.1 – SAM format for the network layer

A.1.2 Packet format

Figure A.2 illustrates a format example of an EAMD security payload (EAMDSP) packet. The packet begins with the EAMDSP header of variable length. Following this field is the payload data, which has a substructure that depends on the choice of encryption algorithm and mode. Following the payload data are the 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 amount of traffic 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.

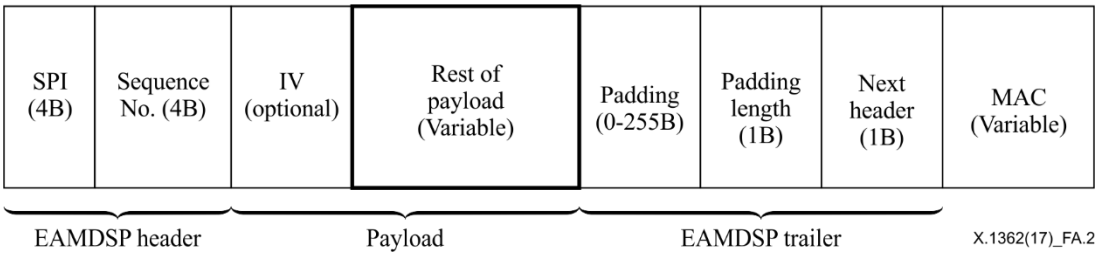

Figure A.2 – Format example of an EAMDSP packet for binding to IPSec ESP protocol

1) Security parameters index (SPI):

SPI is an arbitrary 32-bit value that is used by a receiver to identify SAM to which an incoming packet is bound. The SPI field is mandatory. SPI is carried in the protocol to enable the receiving system to select 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 may also 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 shall 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:

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.

A.1.3 Packet processing

The outbound processing using encryption with associated mask data is 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 clause 6.1.

3) Packet sending:

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

The inbound processing using encryption with associated mask data is as follows:

1) SAM lookup:

   The associated SAM that calls for EAMDSP processing is determined according to some 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 clause 6.1.

A.1.4 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 the encryption algorithm being used according to the mask parameter. For example, in the case of encryption with associated mask data using advanced encryption standard (AES), the payload is split every 128 bits because the block length of AES is 128 bits. The target of the 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 the decryption block. The format of the mask is described in Figures A.3 and A.4. This parameter shows which block should be encrypted or decrypted in case of splitting the payload into the block size of the encryption algorithm being used.

```
MaskFormat ::= SEQUENCE {
  encryptionArea OCTET STRING (SIZE (12))
  reserved OCTET STRING (SIZE (4))
}
```

Figure A.3 – Format of mask

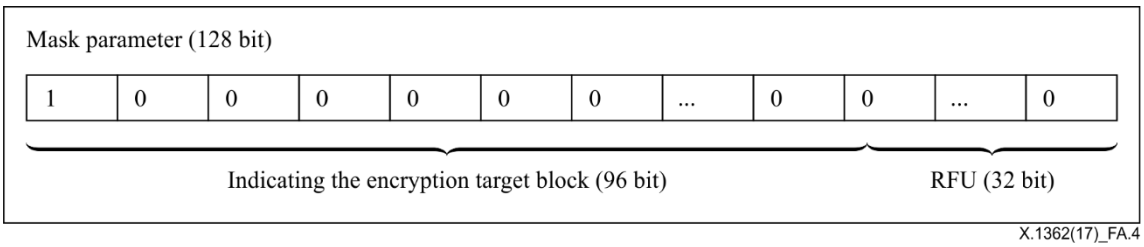

Figure A.4 – Detail format of mask parameter

In this case, the mask parameter means that only the first block is encrypted because the first bit of the mask parameter is true. The encryption of some areas requires changing some bits of the mask parameter from false to true.

A.1.5 Padding algorithm

A padding algorithm may be described as follows:

– Append '0x80' at the end of the payload.

– If the length of the payload is a multiple of block length of encryption algorithm, the padding is finished.

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

Bibliography

[b-ITU-T F.4104] Recommendation ITU-T F.4104/F.744 (2009), *Service description and requirements for ubiquitous sensor network middleware.*

[b-ITU-T X.1311] Recommendation ITU-T X.1311 (2011) | ISO/IEC 29180:2012, *Information technology – Security framework for ubiquitous sensor networks.*

[b-ITU-T X.1312] Recommendation ITU-T X.1312 (2011), *Ubiquitous sensor network middleware security guidelines.*

[b-ITU-T X.1313] Recommendation ITU-T X.1313 (2012), *Security requirements for wireless sensor network routing*.

[b-ITU-T Y.4000] Recommendation ITU-T Y.4000/Y.2060 (2012), *Overview of the Internet of things*.

[b-ITU-T Y.4105] Recommendation ITU-T Y.4105/Y.2221 (2010), *Requirements for support of ubiquitous sensor network (USN) applications and services in the NGN environment.*

[b-ITU-T Y.4109] Recommendation ITU-T Y.4109/Y.2061 (2012), *Requirements for the support of machine-oriented communication applications in the next generation network environment.*

[b-IETF RFC 4301] IETF RFC 4301 (2005), *Security Architecture for the Internet Protocol.*

[b-IETF RFC 5116] IETF RFC 5116 (2008), *An Interface and Algorithms for Authenticated Encryption.*

[b-ISO/IEC 9797] ISO/IEC 9797-1:2011, *Information technology – Security techniques – Message Authentication Codes (MACs) — Part 1: Mechanisms using a block cipher*.

[b-ISO/IEC 18033] ISO/IEC 18033-3:2010, *Information technology – Security techniques – Encryption algorithms – Part 3: Block ciphers.*

[b-ISO/IEC 19772] ISO/IEC 19772:2009, *Information technology – Security techniques – Authenticated encryption*.

[b-ASIACRYPT] Bellare, M., and Namprempre, C. (2000), *Authenticated encryption:* Relations *among notions and analysis of the generic composition paradigm*, in Tatsuaki Okamoto, editor, ASIACRYPT 2000, Vol. 1976 of LNCS, Springer, December, pp. 531-545.

[b-CBCPADD] Vaudenay, S. (2002), *Security Flaws Induced by CBC Padding Applications to SSL, IPSEC, WTLS*, EUROCRYPT 2002.

[b-EUROCRYPT] Namprempre, C., Rogaway, P., and Shrimpton, T. (2014), *Reconsidering generic composition*, in Phong Q. Nguyen and Elisabeth Oswald, editors, EUROCRYPT 2014, Vol. 8441 of LNCS, Springer, May, pp. 257-274.

[b-ZT] Li, Zhang, and Xin, Tong (2013), *Threat Modeling and Countermeasures Study for the Internet of Things*, Journal of Convergence Information Technology (JCIT), Vol. 8, No. 5, March.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks, open system communications and security**
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2017