Suggestion for amendment to IEEE Std 802.1AE— Cover pages

Suggestion for amendment to IEEE Std 802.1AE
April 9, 2025

(Amendment of IEEE Std 802.1AE™-2018 as amended by IEEE Std 802.1AE-2018/Cor1-2020 and IEEE Std 802.1AEdk™-2023)

Media Access Control (MAC) Security

Amendment 5:

Ascon Cipher Suite

Individual contribution

This document is a detailed analysis of the changes that, in my current opinion, that would be required to add an Ascon based Cipher Suite to IEEE Std 802.1AE Media Access Control (MAC) Security [MACsec]. It is intended to facilitate discussion of a proposed PAR (Project Authorization Request).

Some fairly extensive footnotes commenting on various aspects. They are not intended for inclusion in any published amendment, and are distinguished by the Cyan background to this present text.

Mick Seaman

The Introduction to the current draft provides background, and may be updated if there is more than one draft of this document.

The text proper of this draft begins with the Title page.

Introduction to the current draft

Background

The Competition for Authenticated Encryption: Security, Applicability, and Robustness (CAESAR) was announced in January 2013 and we (the IEEE 802.1 Security Task Group) have been following it intermittently since then as a potential source for a MACsec Cipher Suite for constrained devices. The final CAESAR portfolio included the Ascon authenticated encryption scheme. Ascon provides AEAD (Authenticated Encryption with Associated Data) as needed for MACsec (Destination and Source MAC Addresses and the SecTAG remain in the clear, while following data can be encrypted, with the whole frame from DA through to the last octet of the User Data being integrity protected). NIST selected Ascon for lightweight cryptography to protect small devices.[1], [2]

NIST SP 800-232 (Initial Public Draft) ‘Ascon-Based Lightweight Cryptography Standards for Constrained Devices: Authenticated Encryption, Hash, and Extendable Output Functions’ [3] was published November 2024, with the public comment period closing February 2025.[4]

Draft for a draft amendment

This draft is intended to be a fairly complete set of suggested changes for an amendment to IEEE Std 802.1AE to include an Ascon (NIST SP 800-232) based Cipher Suite. Clearly an IEEE project to standardize such an amendment could not complete until the NIST document is finalized. However, although the work in adding such a Cipher Suite by reference to the NIST document might appear trivial, the devil is always in the details and the process itself can take time. It seems wise to start the effort now to reduce delay following NIST finalization.

What’s missing [5]

Additions and changes to the PICS (Annex A) are required.

Ascon-XPN-128 additions to Annex C ‘Test Vectors’ are neeeded. For clarity as to what contributions to the nonce are required, the received SecTAG (in wire order) and the prior extended PN should be included for this Cipher Suite.

[1] https://csrc.nist.gov/news/2023/lightweight-cryptography-nist-selects-ascon

[2] https://www.nist.gov/news-events/news/2023/02/nist-selects-lightweight-cryptography-algorithms-protect-small-devices

[3] https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-232.ipd.pdf

[4] https://csrc.nist.gov/pubs/sp/800/232/ipd

[5] Apart from 802.1 agreement to pursue this work, an approved PAR, and anything that comes up in the course of work.

This page intentionally left blank

Suggestion for amendment to IEEE Std 802.1AE
April 9, 2025
(Amendment of IEEE Std 802.1AE™-2018 as amended by IEEE Std 802.1AE-2018/Cor1-2020 and IEEE Std 802.1AEdk™-2023)

Media Access Control (MAC) Security

Amendment 5: Ascon Cipher Suite

Individual contribution — Mick Seaman

Abstract: This standard specifies how all or part of a network can be secured transparently to peer protocol entities that use the MAC Service provided by IEEE 802® LANs to communicate. MAC security (MACsec) provides connectionless user data confidentiality, frame data integrity, and data origin authenticity.

Keywords: Ascon, authorized port, bridged networks, confidentiality, data origin authenticity, IEEE 802.1AE™, IEEE 802.1AEbn™, IEEE 802.1AEbw™, IEEE 802.1AEcg™, IEEE 802.1AEdk™, GCM, integrity, LANs, local area networks, MAC Bridges, MAC security, MAC Service, MANs, metropolitan area networks, port-based network access control, secure association, security, transparent bridging

Introduction

This introduction is not part of IEEE Std 802.1AExx™-202X, IEEE Standard for Local and metropolitan area networks—Media Access Control (MAC) Security—

The first edition of IEEE Std 802.1AE was published in 2006.

IEEE Std 802.1AEbn™-2011 added the option of using the GCM-AES-256 Cipher Suite.

IEEE Std 802.1AEbw™-2013 added the GCM-AES-XPN-128 and GCM-AES-XPN-256 Cipher Suites. These extended packet numbering (XPN) Cipher Suites allow more than 2^{32} frames to be protected with a single Secure Association Key (SAK) and so ease the timeliness requirements on key agreement protocols for very high speed (100 Gb/s plus) operation.

IEEE Std 802.1AEcg™-2017 specified Ethernet Data Encryption devices (EDEs) that provide transparent secure connectivity while supporting provider network service selection and provider backbone network selection as specified in IEEE Std 802.1Q™.

The IEEE Std 802.1AE-2018™ revision incorporated the text of IEEE Std 802.1AE-2006, amendments IEEE Std 802.1AEbn-2011, IEEE Std 802.1AEbw-2013, and IEEE Std 802.1AEcg-2017.

The IEEE Std 802.1AE-2018/Cor 1-2020 corrigendum corrected an unintentional change to Figure 9-4, replacing it with Figure 9-4 of IEEE_Std_802.1AE-2006.

IEEE Std 802.1AEdk™-2023 specified MAC privacy protection, and added YANG modules for MAC Security and MAC Privacy protection.

IEEE Std 802.1AExx-202X added the Ascon-XPN-128 Cipher Suite.

Relationship between IEEE Std 802.1AE and other IEEE Std 802® standards

IEEE Std 802.1X™-2010 specifies Port-based Network Access Control, provides a means of authenticating and authorizing devices attached to a LAN, and includes the MACsec Key Agreement protocol (MKA) necessary to make use of the MAC Security protocol (MACsec) specified in IEEE Std 802.1AE.

MACsec is not intended as a substitute for the security mechanisms specified by IEEE Std 802.11™ Wireless LAN Medium Access Control. That standard also uses IEEE Std 802.1X, thus facilitating the use of a common authentication and authorization framework for LAN media to which this standard applies and for Wireless LANs.

Contents

5. Conformance 8
5.1 MAC Security Entity requirements 8

10. Principles of MAC Security Entity (SecY) operation 9
10.7.28 SAK creation 9

14. Cipher Suites 11
14.1 Cipher Suite conformance 11
14.1.1 Conformance with Cipher Suite variance 12
14.9 Ascon-XPN-128 12
14.9.1 Packet Number limitation 13
14.9.2 Endian issues 14
14.9.3 Nonce construction 15

Figures

Figure 14-1 Initial MPDU associated data Ascon-AEAD128 block.. 14
Figure 14-2 Ascon-XPN-128 Nonce construction 15

Tables

Table 10-1 MKA Salt construction (examples) 10
Table 14-1 MACsec Cipher Suites.......... 11
Table 14-2 Nonce construction (example) 16

IEEE Standard for
Local and Metropolitan Area Networks—

Media Access Control (MAC) Security

Amendment 5:
Ascon Cipher Suites

[This amendment is based on IEEE Std 802.1AE™-2018 as amended by IEEE Std 802.1AE-2018/Cor1-2020 and IEEE Std 802.1AEdk™-2023]

NOTE—The editing instructions contained in this amendment define how to merge the material contained therein into the existing base standard and its amendments to form the comprehensive standard.

The editing instructions are shown in ***bold italics***. Four editing instructions are used: change, delete, insert, and replace. ***Change*** is used to make corrections in existing text or tables. The editing instruction specifies the location of the change and describes what is being changed by using ~~strikethrough~~ (to remove old material) and underscore (to add new material). ***Delete*** removes existing material. ***Insert*** adds new material without disturbing the existing material. Deletions and insertions may require renumbering. If so, renumbering instructions are given in the editing instruction. ***Replace*** is used to make changes in figures or equations by removing the existing figure or equation and replacing it with a new one. Editing instructions, change markings, and this note will not be carried over into future editions because the changes will be incorporated into the base standard.

5. Conformance

5.1 MAC Security Entity requirements

Change item k) of 5.3 as follows:

k) Support Integrity Protection using ~~the Default Cipher Suite~~ either or both of the mandatory Cipher Suites specified in Clause 14.[6]

[6] To date we have tried to maximize interoperability by requiring a single Default Cipher Suite (GCM-AES-128) while providing fully conformant options for others (notably the GCM-AES based XPN Cipher Suites). Cost considerations throughout the value change, from development to deployment, appeared to support that approach even as alternative Cipher Suites with different cost and performamce characteristics have become available. Lightweight cryptography cost opportunities and requirements have stretched that single fous to breaking point, so this amendment would no longer require GCM for conformamce. Requiring an inadequately performing software based GCM implementation on constrained devices for conformance would serve no purpose, and we could expect any such requirment to be almost universally ignore. I have simply added choice in this amendment, however there might be a way/claim label to make it clear as to which Cipher Suite(s) are part of the conformant implementation.

10. Principles of MAC Security Entity (SecY) operation

10.7.28 SAK creation

Change the text of 10.7.28 and insert Table 10-3 as follows, renumbering subsequent Tables:

An SAK is installed, i.e., an instance of the Current Cipher Suite for a given SAK is created, on request from the KaY with the following parameters:

a) The SAK value

b) keyIdentifier, used by network management to reference the key

c) transmit, True if the key is to be installed for transmission

d) receive, True if the key is to be installed for reception

and, if the Current Cipher Suite uses extended packet numbering, the following parameter:

e) a Salt (McGrew [B17])~~, a 96-bit parameter provided to the Current Cipher Suite for subsequent protection and validation operations~~

If a MACsec Key Agreement protocol (MKA) Key Server does not distribute an ~~include~~ explicit ~~parameters for distributing a~~ Salt, ~~E~~each KaY computes this parameter ~~as follows~~.

GCM-AES-XPN-128 (14.7) and GCM-AES-XPN-256 (14.8) use a 96-bit Salt. The 64 least significant bits ~~of the Salt~~ are the 64 least significant bits of the MKA Key Server's Member Identifier (MI), the 16 next most significant bits ~~of the Salt~~ comprise the exclusive-or of the 16 next most significant bits of that MI with the 16 most significant bits of the 32-bit MKA Key Number (KN), and the 16 most significant bits ~~of the Salt~~ comprise the exclusive-or of the 16 most significant bits of that MI with the 16 least significant bits of the KN.

Ascon-XPN-128 (14.9) uses a 128-bit Salt, computed as follows:

— The 48 least significant bits are the 48 least significant bits of the MKA Key Server's Member Identifier (MI).

— The 16 next most significant bits are the exclusive-or of the 16 next most significant bits of that MI with the 16 least significant bits of the 32-bit MKA Key Number (KN).

— The 32 next most significant bits are the 32 most significant bits of that MI.

— The 16 next most significant bits are the 16 least significant bits of that MI.

— The 8 next most significant bits are the 8 most significant bits of the 24 least significant bits of the Key Server's MI exclusive or'd with 8 most significant bits of the KN.

— The 8 most significant bits of the Salt are the 8 most significant bits of the 32 least significant bits of that MI exclusive or'd with the 8 least significant bits of the 16 most significant bits of the KN.

~~This way~~ These ways of obtaining a Salt ~~is~~ are not necessarily applicable to any other key agreement protocol. Table 10-1 provides examples.[7]

[7] Both the existing96-bit Salt (specified for use with the GCM XPN Cipher Suites) and the 128-bit Salt specified for use with Ascon-XPN-128 have the property of creating different nonce spaces for successively numbered (KN values) SAKs. They differ because the GCM XPN Cipher Suites use a 96-bit nonce (IV), thus requiring the allocation of an SSCI to allocate the nonce space to CA participants, whereas Ascon uses a 128-bit nonce, allowing the full SCI to be part of the nonce (and thus not duplicating nonce space use for a given SAK for participants anywhere that have non-colliding universally allocated MAC addresses). See 14.9 for further detail.

Table 10-1—MKA Salt construction (examples)

Key Number (KN)	1234 5678$_{16}$[a]
Key Server MI	1122 3344 5566 7788 99AA BBCC$_{16}$
96-bit Salt	475A 2170 5566 7788 99AA BBCC$_{16}$
128-bit Salt	ADB8 BBCC 1122 3344 031E 7788 99AA BBCC$_{16}$
Key Number (KN)	0001 2853$_{16}$
Key Server MI	E630 E81A 48DE 85B4 6A21 C66F$_{16}$
96-bit Salt	CE63 E81B 48DE 85B4 6A21 C66F$_{16}$
128-bit Salt	6B21 C66F E630 E81A 608D 85B4 6A21 C66F$_{16}$

[a] All the entries in this table are hexadecimal integers, with the most significant digits to the left. Spaces have been inserted, breaking each integer into 16-bit blocks, for ease of comprehension.

14. Cipher Suites

14.1 Cipher Suite conformance

Change the first two paragraphs of 14.4, and the NOTE following, as follows:

An implementation of MACsec that claims full conformance to this standard shall implement either or both of the mandatory Cipher Suites in Table 14-1, may implement one or more of the associated Optional Cipher Suites in the table, and shall not implement any other Cipher Suite. Every conformant implementation shall include at least one Cipher Suite that does not encrypt User Data.

Change Table 14-1 adding a new row as follows (all rows shown):

Table 14-1—MACsec Cipher Suites

Cipher Suite Identifier	**Cipher Suite Name**	**Services provided**		**Mandatory/Optional**	**Defining Clause**
		Integrity without confidentiality	**Integrity and confidentiality**		
00-80-C2-00-01-00-00-01	GCM-AES-128	Yes	Yes	Mandatory [a]	14.5
00-80-C2-00-01-00-00-02	GCM-AES-256	Yes	Yes	Optional [b]	14.6
00-80-C2-00-01-00-00-03	GCM-AES-XPN-128	Yes	Yes	Optional [b]	14.7
00-80-C2-00-01-00-00-04	GCM-AES-XPN-256	Yes	Yes	Optional [b]	14.8
00-80-C2-00-01-00-00-10[c]	Ascon-XPN-128	Yes	Yes	Mandatory [a]	14.9

[a] At least one of the Mandatory Cipher Suites is to be implemented by any conformant implementation.

[b] May be implemented if, and only if, GCM-AES-128 (the Default Cipher Suite) is implemented.

[c] Assigned number deliberately leaving number space, shoulld any further GCM-AES or other AES variant be required. Since there is more than enough number space available it does not seem necessary to delay assignment of an identifier and require the choice of another for interoperability testing. A different or succession of different identifiers could be assigned if there is any material change in Cipher Suite specification.

Table 14-1 assigns a Cipher Suite reference number for use in protocol identification within a MACsec context, provides a short name for use in this standard, indicates the type of cryptographic algorithm used and the security services provided,~~ specifies whether the Cipher Suite is mandatory or optional for conformance to this standard,~~ and references the clause of this standard that provides the definitive description of the Cipher Suite.

NOTE—In IEEE Std 802.1AE-2006 (the first edition of this standard), the Cipher Suite Identifier for GCM-AES-128 was incorrectly shown as 00-80-02-00-01-00-00-01 in Table 14-1. Prior to the inclusion of GCM-AES-256, GCM-AES-128 was the only conformant Cipher Suite. IEEE Std 802.1X uses a reserved encoding for the Default Cipher Suite rather than the Cipher Suite Identifier to identify GCM-AES-128.

14.1.1 Conformance with Cipher Suite variance

Change the first paragraph of 14.4.1 as follows:

An implementation of MACsec that claims conformance to this standard with Cipher Suite variance, shall implement either or both of the mandatory Cipher Suites in Table 14-1, may implement one or more of the associated optional Cipher Suites in Table 14-1, and may implement alternate Cipher Suites that meet the requirements of 14.2 and 14.3, and the following guidelines, and shall not implement any other Cipher Suite, or other combination of cryptographic algorithms and parameters.

Insert 14.9 as follows:

14.9 Ascon-XPN-128[8]

The Ascon-XPN-128 Cipher Suite protects each frame using nonce-based authentication encryption with associated data as specified in this clause by reference to the Ascon-AEAD128 algorithm and the associated terms *K*, *N*, *A*, *P*, *C*, *T* specified in NIST SP 800-232.

K is the 128-bit SAK.

N is the 128-bit nonce, and is constructed as specified in 14.9.3.

T is the 128-bit ICV.

When this Cipher Suite is used for Integrity Protection

- *A* is the Destination MAC Address, Source MAC Address, and the first four octets of the SecTAG and the octets of the User Data concatenated in that order.
- *P* and *C* are null.
- The Secure Data is the octets of the User Data, without modification.

When this Cipher Suite is used for Confidentiality Protection

- *A* is the Destination MAC Address, Source MAC Address, and the first four octets of the SecTAG concatenated in that order.
- *P* is the octets of the User Data.
- The Secure Data is *C*.

This Cipher Suite does not provide Confidentiality Protection with a confidentiality offset.

NOTE—The specification of associated data, A, for this Cipher Suite differs from that for the GCM Cipher Suites.[9]

[8] The rationale for making this an XPN Cipher Suite, and limits on the PN range to be used while respecting the NIST SP 800-232 limit on the number of Ascon blocks to be protected using a single key are provided in text and in editor's comments below.

[9] All the octets of the SecTAG are included in *A* for GCM, i.e., the four octets of the 32-bit field and the SCI (if present in the SecTAG) were also included. This was done to simplify hardware pipeline implementations, which would otherwise have to omit 4 or 12 octets (depending on the ES bit in the SecTAG) from *A* before (if Integrity-only protection was selected) continuing with *A*, or (if Confidentiality was being provided) transitioning to *P*. Target implementations for Ascon include microcontrollers that are not presumed to couple a flow-through Cipher Suite hardware implementation directly to the 802.3 MAC (media access control) implementation, but implement Ascon in software. In that case excluding all but the first 4 octets of the SecTAG means that for Confidentiality protection *A* comprises a single (128-bit) Ascon block, reducing the number of blocks for small frame processing. The SCI field(s) omitted from *A* are naturally protected by their inclusion in the nonce, *N*. As a minor implementation note it can be advantageous to allocate memory buffers for received frames so the following Ciphertext, *C*, (for Confidentiality protection) or the remaining 'associated data' is aligned on word boundaries.

14.9.1 Packet Number limitation

To provide 128-bit security strength, no more than 2^{54} octets (2^{50} blocks) can be protected by a single key (4.4.1 of NIST SP 800-232). This standard limits the Packet Numbers used with this Cipher Suite to the range 1 through $2^{48} – 1$. The 16 most significant bits of the extended Packet Number's possible 64 bits are not used. MAC Operational (6.4) shall be set False if the value of nextPN for secure frame generation is 2^{48} or greater (10.5.2), and no further frames transmitted using that SA. When this Cipher Suite is in use PendingPNExhaustion (8.2.7) takes the value 0xC000 0000 0000. It is the responsibility of key agreement protocol to further limit PN use to satisfy security strength requirements through timely distribution of fresh SAKs.[10]

[10] The maximum PN consumption rate assumed for any Ethernet line rate is that for a frame that, unprotected, comprises 24 or fewer user data octets that is protected using the Short Length field (9.7) and the ES bit (9.5) to avoid any unnecessary padding and inclusion of the SCI in the SecTAG. The protected frame includes the source and destination MAC Addresses (12 octets), the SecTAG (8 octets, including the MACsec Ethertype), and the 16 octet ICV—a total of 36 octets plus user data. When transmitted on Ethernet an FCS of 4 octets is added. If fewer than 24 user data octets are present, the frames is padded to 64 octet prior to transmission with an 8 octet frame and minimum inter-frame gap of 12 octets times, each frame thus occupying 84 octets of wire time. That is the well known maximum rate of 14,880 frames/second for 10 Mb/s Ethernet. At this worst case data rate a 32-bit PN could be exhausted in ~5 minutes @10 Gb/s and ~7 seconds @400 Gb/s. A 48-bit PN could be exhausted in 6 years @1 Gb/s, 22 days @100 Gb/s, and 5 days @400 Gb/s.

Section 4.1.1 'Encryption' of NIST SP 800-232 specifies the parsing of 'associated data' (A) into blocks and (separately) the parsing of 'plaintext' (P) into 128-bit (16-octet blocks) padding any unused bits in each block. When MACsec is providing integrity only protection for the hypothetical minimum frame size, the 12 octets of the MAC DA + SA and the initial 4 octets of the SecTAG are parsed into a single block of 'associated data' (see further on in the main text), and the octets of the user data into an additional single block of 'plaintext'. Assuming the 128-bit security strength requirement of 4.4.1 of NIST SP 800-232 truly applies to block not byte count, a 48-bit PN would be exhausted in half the time need to use 2^{50} blocks . However a more realistic minimum frame (IPv4 and UDP with data) includes at least 31 octets of user data beyond the SecTAG, while a minimum TCP frame pushes the 'plaintext' block count to 3 per frame, so a 48-bit PN is more than adequate (47 bits would have been sufficient for the most outlandish traffic pattern with existing protocols). The worst case data block use rate for is not for minimum size frames, as the wire time for the Ethernet preamble and start of frame delimiter, and for the octets of the SecTAG apart from the first four and the 16 otcet ICV do not contribute to the block count. For maximum size frames (per 802.3 standard, or 'jumbo' 9k frames) these diminish the block usage rate of one per 16 octets of wire time by a few percent at most). With back to back long frames the 2^{50} block pool can be exhausted in 457 years@10 Mb/s, 5 years @1 Gb/s, 167 days @10 Gb/s, 17 days@100 Gb/s, and 100 hours@400 Gb/s. Those figures would be haved for full-duplex Ethernet.

14.9.2 Endian issues

The NIST SP 800-232 specification is little-endian (Appendix A of NIST SP 800-232), whereas the structure and encoding of MACsec Protocol Data Units (MPDUs,) uses big-endian (network byte order) conventions and is Cipher Suite independent. The following provisions of this standard are made explicit to avoid any confusion that might lead to a lack of interoperability between implementations or difficulty in network management. In MPDUs:

— User Data comprises a sequence of octets. When Integrity Protection is provided their transmission order is the same as the order in which they are provided in MSDUs to the transmitting SecY's Controlled Port, and their reception order is the same as the order in which they are delivered to the SecY's transmission port. When presented to or received from the Ascon-AEAD128 protection or validation functions in multi-octet registers or fields, earlier octets are the less bit significant components of those fields.

— Secure Data comprises a sequence of octets. When presented to or received from the Ascon-AEAD128 protection or validation functions in multi-octet registers or fields, earlier octets are the less bit signicant components of those fields.

— The MPDU's Destination and Source MAC Addresses and the MAC Address component of an SCI are sequences of octets, with the octet containing the I/G bit and the U/L bit first in transmission order. When presented to or received from the Ascon-AEAD128 protection or validation functions in multi-octet registers or fields, earlier octets are the less bit significant components of those fields. However when MAC Addresses are to be numerically compared they are treated as binary numbers, with earlier octets being more significant as specified in 9.1.

— The two octet Port Identifiers in SCIs are similarly treated by this Cipher Suite as a sequence of two octets. However when they are to be numerically compared they are treated as binary numbers, with earlier octets being more significant as specified in 9.1.

— The bit significance of the fields of the first 128-bit block of 'associated data', *A*, for both Integrity and Confidentiality protection as presented to the Ascon-AEAD128 protection or validation functions is illustrated in Figure 14-1.

NOTE—The fields in this block are treated as a sequence of octets. While the two octets that compose an EtherType are encoded in network byte order, the difference in imputed octet significance arising from their little-endian treatment for Ascon-AEAD128 processing is not material since both the transmitter, protecting the MPDU, and the receiver, validating it, invoke little-endian processing.

— The lowest 32-bits of the Packet Number (PN) are encoded in the MPDU most significant octet first, so the interpretation, by a network administrator, of the MPDU PN field is consistent with 9.1 and does not depend on knowledge of the current Cipher Suite. If the PN value is to be used in a little-endian register for comparison or arithmetical operations, the octet order of the least significant 32 bits needs to be reversed prior to transmission and on reception. The PN contribution to the Cipher Suite nonce, *N*, does not use transmission order. (14.9.3, Table 14-1).

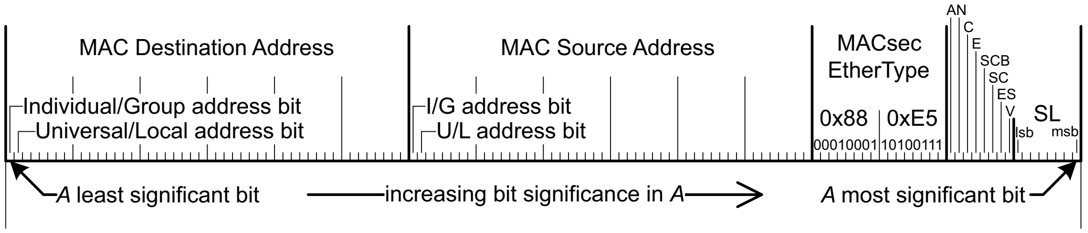

Figure 14-1—Initial MPDU associated data Ascon-AEAD128 block

14.9.3 Nonce construction

The Cipher Suite nonce, *N*, is constructed from the protected frame’s extended Packet Number and the transmitter’s SCI (as encoded in the SecTAG, if present), bit-wise exclusive or’d with a 128-bit Salt (10.7.28) distributed by key agreement protocol to all members of the CA.

NOTE 1—The Salt is not a secret nonce mask as described in 4.2.2 of NIST SP 800-232, however its use does ensure that the nonce is not repeated for up to 2^{16} successive Key Numbers for SAKs distributed by a given Key Server instance (more if no CA participant uses the more significant bits of the Port Identifier as part of its SCI). The Salt also allows fuller use of the nonce space than might otherwise occur, prior to fresh SAK distribution, for lower data rates.

Figure 14-2 shows the PN and SCI contributions to the nonce, and the 128-bit Salt that to be used with MKA unless otherwise specified by IEEE Std 802.1X (10.7.28).

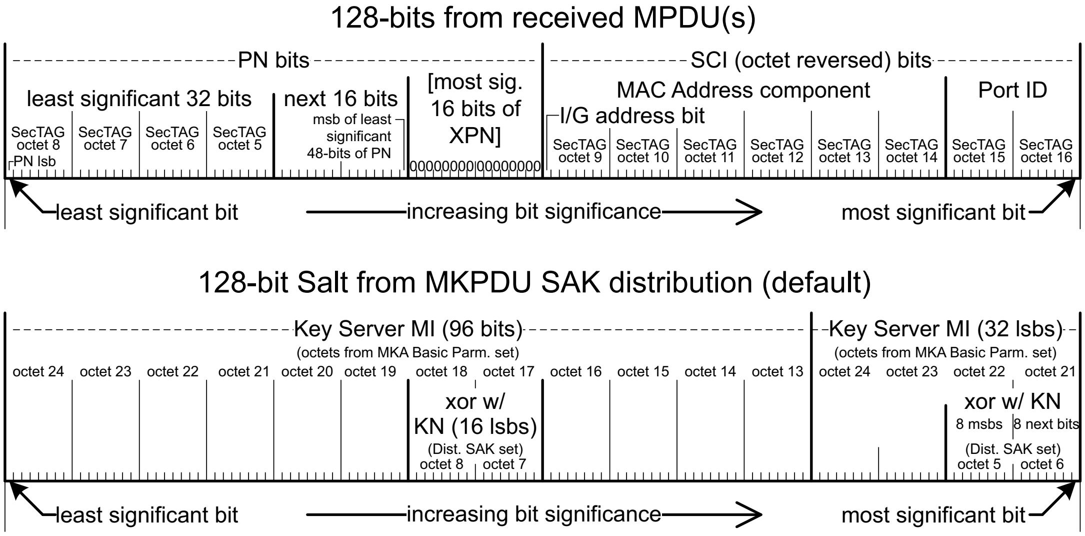

Figure 14-2—Ascon-XPN-128 Nonce construction

The 128 bits contributed to the nonce by the extended Packet Number and the transmitter’s SCI are shown in the upper part of Figure 14-2, and are (reading from right to left, most to least significant) as follows:

— The 8 most significant bits are the 8 least significant bits of the SCI’s Port Identifier.

— The next 8 significant bits are the 8 most significant bits of the SCI’s Port Identifier.

— The next 6 octets comprise the SCI’s MAC Address, with the most significant bits being those of the least significant octet of the SCI when that is treated as a binary number as specified in 9.1.

— The next 16 bits are zero, and correspond to the 16 most significant bits of the extended Packet Number, which are resricted to zero for this Cipher Suite.

— The next 48 bits are the 48 least significant bits of the extended Packet Number.

NOTE 2—The SCI contribution to the nonce corresponds to a little endian interpretation of its reception order (if present in the SecTAG), chosen to avoid the need to reverse the octets on reception for Cipher Suite processing. The PN contribution to the nonce corresponds to the numeric evaluation of the PN field as specified in 9.1 since arithmetic operations (range comparison and increment) are required for PN processing.

Table 14-2 provides an example nonce construction. Sequences of octets are shown as hexadecimal values separated by spaces, in transmission order left (first) to right. Numerical values are shown in hexadecimal, most significant digits to the left, with leading zero padding as appropriate to the notional size of the integer, with spaces breaking integers into 16-bit blocks for ease of comprehension..

Table 14-2—Nonce construction (example)

Frame fields	Value in protected frame [a]
MAC DA	`E2 01 06 D7 CD 0D`
MAC SA	`F0 76 1E 8D CD 3D`
MACsec EtherType	`88 E5`
TCI and AN	`2C` [b]
SL	`00` [c]
PN field octets	`76D4 57ED` [e]
SCI	`68 F2 E7 76 96 CE 00 01` [d]
SecY parameters	**Value (applicable to protected frame) [e]**
lowest acceptable PN	`0000 0025 76D4 57DD`
PN	`0000 0025 76D4 57ED`
Key Number (KN)	`0001 2853`
Key Server MI	`E630 E81A 48DE 85B4 6A21 C66F`
Salt	`6B21 C66F E630 E81A 608D 85B4 6A21 C66F`
Nonce prior to Salt	`0100 CE96 76E7 F268 0000 0025 76D4 57ED`
Nonce	`6A21 08F9 90D7 1A72 608D 8591 1CF5 9182`

[a] Octets (apart from PN) in transmission order, left to right, hexadecimal.

[b] SCI included in SecTAG, Confidentiality protection, AN = 0.

[c] Not short, frame contains 48 or more octets of User Data.

[d] Comprising a MAC Address with a Port Identfier of 0001

[e] Integers, hexadecimal, most significant digits to the left, leading zero padding to notional size, spaces added to break into 16-bit blocks do not denote transmission order.