IEC 62056-8-8

Edition 1.0 2020-04

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

Electricity metering data exchange – The DLMS/COSEM suite – Part 8-8: Communication profile for ISO/IEC 14908 series networks

Échange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 8-8: Profil de communication pour réseaux de la série ISO/IEC 14908

IEC 62056-8-8:2020-04(en-fr)

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
info@iec.ch
www.iec.ch

IEC 62056-8-8

Edition 1.0 2020-04

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

Electricity metering data exchange – The DLMS/COSEM suite – Part 8-8: Communication profile for ISO/IEC 14908 series networks

Échange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 8-8: Profil de communication pour réseaux de la série ISO/IEC 14908

INTERNATIONAL ELECTROTECHNICAL COMMISSION

COMMISSION ELECTROTECHNIQUE INTERNATIONALE

ICS 17.220; 35.110; 91.140.50

ISBN 978-2-8322-8082-9

CONTENTS

FOREWORD 7
INTRODUCTION 9
1 Scope 10
2 Normative references 10
3 Terms, definitions and abbreviated terms 11
3.1 Terms and definitions 11
3.2 Abbreviated terms 12
4 Targeted communication environments 13
5 Use of the communication layers for this profile 13
5.1 Information related to the use of the standard specifying the lower layers 13
5.2 The structure of the communication profile 14
5.3 Lower layers and their use 14
5.3.1 Overview 14
5.3.2 Physical layer 15
5.3.3 Data link layer 15
5.3.4 Network layer 15
5.3.5 Transport layer 15
5.4 Service mapping and Adaptation layer 16
5.4.1 Overview 16
5.4.2 Adaptation layer 16
5.5 Registration and connection management 27
6 Identification and addressing schemes 27
7 Specific considerations for the application layer services 28
7.1 Overview 28
7.2 Application Association establishment and release: ACSE services 28
7.2.1 Application association establishment 28
7.2.2 Application association release 29
7.2.3 COSEM-OPEN and COSEM-RELEASE service parameters 29
7.3 xDLMS services 30
7.4 Security mechanisms 30
7.4.1 DLMS/COSEM security 30
7.4.2 Adaptation layer security 30
7.4.3 Lower layers security 30
7.5 Transferring long application messages 30
7.6 Media access, bandwidth and timing considerations 30
7.7 Other considerations 30
8 Communication configuration and management 31
9 The COSEM Application Process 31
10 Additional considerations for the use of this profile 31
Annex A (informative) Examples 32
Annex B (normative) Interface classes 33
B.1 Interface classes 33
B.1.1 ISO/IEC 14908 Identification 33
B.1.2 ISO/IEC 14908 Protocol setup 33
B.1.3 ISO/IEC 14908 Protocol status 34

B.1.4 ISO/IEC 14908 Diagnostic 35
B.2 Relation to OBIS 37
Annex C (informative) Implementation guide and migration path 38
C.1 General 38
C.2 Client-server model 38
C.2.1 General 38
C.2.2 Clients 38
C.2.3 Servers 39
C.2.4 Application associations 39
C.3 The COSEM objects 40
C.3.1 COSEM object model 40
C.3.2 Referencing methods 41
C.3.3 The object_list 42
C.3.4 Mandatory COSEM objects 42
C.3.5 Application specific objects 42
C.3.6 Utility table objects 42
Annex D (informative) OSGP-AES-128-PSK security suite 46
D.1 General 46
D.2 Background 47
D.2.1 Overview 47
D.2.2 System Assumptions 47
D.2.3 Threat Model 47
D.2.4 Design goals 48
D.2.5 Inspiration 48
D.3 Terms and definition, abbreviated terms and notation 48
D.3.1 Terms and definitions 48
D.3.2 Abbreviated terms 50
D.3.3 Notation 50
D.3.4 Other conventions 50
D.4 Cryptographic Primitives 51
D.4.1 General 51
D.4.2 CMAC 51
D.4.3 CCM 51
D.5 Cryptographic Functions 52
D.5.1 Overview 52
D.5.2 OSGP_KDF: Key Derivation Function 52
D.5.3 OSGP_MAC: Message Authentication Code Function 52
D.5.4 OSGP_MAC_VERIFY: Message Authentication Code Verification Function 53
D.5.5 OSGP_AE/OSGP_AD: Authenticated Encryption/Decryption Function 53
D.5.6 OSGP_CSPRNG: Cryptographically Secure Pseudo Random Number Generator 55
D.6 Keys 56
D.7 Secure channel initialization 58
D.7.1 Overview 58
D.7.2 Secure Channel State (CryptoContext) 58
D.7.3 Flow 58
D.7.4 Security Suite Negotiation 63
D.7.5 LNAP Commissioning 64

D.7.6 Error Handling and Intrusion Detection 64
D.7.7 Messages 64
D.8 Secure Channel Communication 67
D.8.1 General 67
D.8.2 The General Process 68
D.8.3 Unicast communication 69
D.8.4 Broadcast Communication 73
D.9 Firmware Downloading 75
D.10 Key Management 76
D.10.1 Renewing the Short-term, Meter-unique Keys 76
D.10.2 Renewing the Short-term, Domain-unique Keys 76
D.10.3 UpdateDomainKeysRequest 77
D.10.4 UpdateDomainKeysResponse 77
D.11 Updating the Long-term, Meter-unique Keys 77
D.11.1 General 77
D.11.2 Key Validity Periods 78
D.12 Error Messages 78
D.12.1 Overview 78
D.12.2 AuthenticationFailure 78
D.12.3 SequenceError 78
D.13 Security considerations 78
D.13.1 Reasoning 78
D.13.2 Recommendation and Guidance for Implementers 81
Annex E (normative) Repeating mechanism 83
E.1 Overview 83
E.2 Terms and definitions 83
E.3 Protocol specification 84
E.3.1 Overview 84
E.3.2 Addressing 84
E.3.3 Service types 84
E.3.4 Timers 84
E.3.5 Request flow 85
E.3.6 Response flow 85
E.3.7 Authentication 85
E.3.8 Examples 86
E.3.9 Broadcast 87
E.4 Downlink frame format 87
E.4.1 Overview 87
E.4.2 Proxy parameters of request frame 88
E.5 Uplink Frame format 92
E.5.1 Overview 92
E.5.2 Proxy success 92
E.5.3 Repeating Failure 93
E.5.4 Authentication failure 93
Annex F (informative) ISO/IEC 14908-3 Registration and monitoring of LNAPs 94
F.1 Scope 94
F.2 Terms and definitions 94
F.3 Metering device lifecycle 94
F.4 ATM Protocol 95

F.4.1 General 95
F.4.2 ATM roles 95
F.4.3 Automatic discovery 95
F.4.4 Discovery domain 96
F.4.5 ATM messages 96
F.5 Commissioning 100
F.5.1 Overview 100
F.5.2 Commissioning operations 100
Bibliography 101

Figure 1 – Functional reference architecture 13
Figure 2 – Structure of the communication profile 14
Figure 3 – Adaptation layer structure 17
Figure 4 – Unprotected Adaptation layer PDU structure 18
Figure 5 – Protected Adaptation layer PDU structure 19
Figure 6 – Adaptation layer services 20
Figure 7 – Client and server ADP_DATA services 24
Figure D.1 – Key hierarchy diagram 56
Figure D.2 – Secure channel establishment 59
Figure D.3 – Challenge request 60
Figure D.4 – Challenge response 60
Figure D.5 – Commission Request 61
Figure D.6 – Commission response 62
Figure D.7 – Crypto context establishment 67
Figure D.8 – Authenticate encryption mechanism 68
Figure D.9 – Authenticated decryption 69
Figure E.1 – Unacknowledged message exchange 86
Figure E.2 – Request / Response message 86
Figure E.3 – Error during request transmission 86
Figure E.4 – Error during response transmission 87
Figure E.5 – Proxy Repeater proxy message 87
Figure E.6 – Proxy Agent proxy message format 87
Figure E.7 – Proxy Target message format 88
Figure E.8 – Normal ProxyTrailer 90
Figure E.9 – Alternate Key Proxy Trailer 90
Figure E.10 – Proxy SICB field structure 90
Figure E.11 – Alternate Key format 91
Figure E.12 – ProxySubnetNode Address (Type1) 92
Figure E.13 – ProxyUniqueNode_ID Address (Type 2) 92
Figure E.14 – ProxyBroadcast Address (Type 3) 92
Figure E.15 – ProxySubnetNode Compact Address (Type 5 and 7) 92
Figure E.16 – ProxyUniqueNodeCompact Address (Type 6) 92

Table 1 – Control field 18

Table 2 – Client and Server SAP 28
Table 3 – Application associations and data exchange in the DLMS/COSEM over ISO/IEC14908 profile 29
Table C.1 – Pre-established association parameters 40
Table C.2 – base_name range for tables categories 43
Table C.3 – base_names for standard tables BT00 to BT79 44
Table D.1 – Security logical layers 46
Table D.2 – Key type and their scope 57
Table D.3 – Challenge request 65
Table D.4 – Challenge response 65
Table D.5 – Commission request 66
Table D.6 – Commission response structure 67
Table D.7 – Request message 70
Table D.8 – response message structure 70
Table D.9 – Nonce construction 70
Table D.10 – Associated Data construction 71
Table D.11 – Broadcast message 74
Table D.12 – Broadcast nonce 74
Table D.13 – Broadcast Associated Data 74
Table D.14 – Update Domain Keys request 77
Table D.15 – Update Domain Keys response 77
Table D.16 – AuthenticationFailure message 78
Table D.17 – SequenceError message 78
Table D.18 – Example of risks 82
Table E.1 – ProxyHeader format 88
Table E.2 – ProxyDetailedAddress format 89
Table E.3 – ProxyUniformAddress format 89
Table E.4 – ProxyTx control format 89
Table E.5 – Address Type values 90
Table E.6 – Service types 91
Table E.7 – Mode types 91
Table E.8 – Repeating success response 93
Table E.9 – Repeating Failure 93
Table E.10 – Authentication failure code 93
Table F.1 – ATM Message header structure 97
Table F.2 – ATM payloads 97
Table F.3 – ATM-query-ID request structure 98
Table F.4 – ATM-query-ID response structure 98
Table F.5 – ATM-respond-to-query request structure 99
Table F.6 – ATM respond-to-query response structure 99
Table F.7 – Flags 100

INTERNATIONAL ELECTROTECHNICAL COMMISSION

_____

ELECTRICITY METERING DATA EXCHANGE – THE DLMS/COSEM SUITE –

Part 8-8: Communication profile for ISO/IEC 14908 series networks

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62056-8-8 has been prepared by IEC technical committee 13, Electrical energy measurement and control.

The text of this International Standard is based on the following documents:

CDV	Report on voting
13/1783/CDV	13/1792/RVC

Full information on the voting for the approval of this International Standard can be found in the report on voting indicated in the above table.

This document has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts in the IEC 62056 series, published under the general title *Electricity metering data exange – The DLMS/COSEM suite*, can be found on the IEC website.

The committee has decided that the contents of this document will remain unchanged until the stability date indicated on the IEC website under "http://webstore.iec.ch" in the data related to the specific document. At this date, the document will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

INTRODUCTION

The IEC 62056 DLMS/COSEM suite provides specific communication profile standards for communication media relevant for smart metering.

Such communication profile standards specify how the COSEM data model and the DLMS/COSEM application layer can be used on the lower communication media-specific protocol layers.

Communication profile standards refer to communication standards that are part of the IEC 62056 DLMS/COSEM suite or to any other open communication standard.

This International Standard specifies DLMS/COSEM communication profile using ISO/IEC 14908-1:2012, *Information technology – Control network protocol – Part 1: Protocol stack* and ISO/IEC 14908-3:2012, *Information technology – Control network protocol – Part 3: Power line channel specification.* It applies for devices installed on the neighbourhood area network.

It follows the rules defined in IEC 62056-5-3:2017, Annex A, and in IEC 62056-1-0, and IEC TS 62056-1-1 for its structure.

ELECTRICITY METERING DATA EXCHANGE – THE DLMS/COSEM SUITE –

Part 8-8: Communication profile for ISO/IEC 14908 series networks

1 Scope

This part of IEC 62056 describes how the DLMS/COSEM Application layer and the COSEM object model as specified in IEC 62056-5-3:2017, IEC 62056-6-1:2017 and IEC 62056-6-2:2017 can be used over the lower layers specified in the IEC 14908 series, forming a DLMS/COSEM ISO/IEC 14908 communication profile.

This document is part of the IEC 62056 series. Its structure follows IEC 62056-1-0 and IEC TS 62056-1-1.

Annex A (informative) provides examples of representative instances of data exchange.

NOTE This Annex A is included and referenced for consistency with other parts of the IEC 62056 suite, but it is empty.

Annex B (normative) defines COSEM interface classes and related OBIS codes for setting up and managing the DLMS/COSEM communication profile for IEC 14908 networks. These interface classes and OBIS codes will be moved later to IEC 62056-6-2 and IEC 62056-6-1.

Annex C (informative) provides an implementation guide and specifies a migration path from Utility Tables based applications to DLMS/COSEM based applications.

Annex D (informative) specifies the OSGP-AES-128-PSK security suite for optional use on the adaptation layer level.

Annex E (normative) specifies the repeating mechanism over the ISO 14908-3 Power Line Channel network.

Annex F (informative) specifies ISO/IEC 14908-3 Registration and monitoring of LNAPs.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 62056-5-3:2017, *Electricity metering data exchange – The DLMS/COSEM suite – Part 5-3: DLMS/COSEM application layer*

IEC 62056-6-1:2017, *Electricity metering data exchange – The DLMS/COSEM suite – Part 6-1: Object Identification System (OBIS)*

IEC 62056-6-2:2017, *Electricity metering data exchange – The DLMS/COSEM suite – Part 6-2: COSEM interface classes*

ISO/IEC 14908-1:2012, *Information technology – Control network protocol – Part 1: Protocol stack*

ISO/IEC 14908-3:2012, *Information technology – Control network protocol – Part 3: Power line channel specification*

EN 50065-1, *Signalling on low-voltage electrical installations in the frequency range 3 kHz to 148,5 kHz – Part 1: General requirements, frequency bands and electromagnetic disturbances*

3 Terms, definitions and abbreviated terms

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at http://www.electropedia.org/
- ISO Online browsing platform: available at http://www.iso.org/obp

NOTE All the messages in this document use the big endian format

3.1 Terms and definitions

3.1.1
domain
logical network that is a unit for addressing

Note 1 to entry: Subnet (see below) and node addresses are assigned by the administrator responsible for the domain, and they have meaning only in the context of that domain.

Note 2 to entry: All nodes belongs to the same domain to be able to address each other.

3.1.2
node
abstraction for a physical node that represents the highest degree of address resolvability on a network

Note 1 to entry: A node is identified (addressed) within a subnet by its (logical) node identifier called Node_ID. A physical node may belong to more than one subnet; when it does, it is assigned one (logical) node number for each subnet to which it belongs. A physical node may belong to at most two subnets; these subnets are parts of different domains. A node may also be identified (absolutely) within a network by its Unique_Node_ID which is immutable.

3.1.3
subnet
set of nodes accessible through the same link layer protocol

Note 1 to entry: In a logical address, a subnet is identified by a Subnet_ID.

3.1.4
transaction
sequence of messages that are correlated

Note 1 to entry: For example, a request and the response to the request are all part of a single transaction. A transaction succeeds when all the expected messages from every node involved in the transaction are received at least once. A transaction fails if any of the expected messages within the transaction are not received.

3.2 Abbreviated terms

Abbreviated term	Meaning
AA	Application Association
AARE	Application Association Response
AARQ	Application Association Request
ACSE	Association Control Service Element
ADD	Automated Device Discovery
AdpPDU	Adaptation layer PDU
AL	Application layer
AP	Application Process
APDU	Application Protocol Data Unit
ASE	Application Service Element
ATM	Automated Topology Management
BPSK	Binary Phase Shift Keyed
COSEM	Companion Specification of Energy Metering
CRC	Cyclic Redundancy Code
CSMA	Carrier Sense Medium Access
DC	Data Concentrator
DLPDU	Data Link Protocol Data Unit
DLMS	Device Language Message Specification
FCM	Fast Commission Message
KMS	Key Management System
HES	Head End System
LNAP	Local Network Access Points
MAC	Medium Access Control (sublayer)
MAC	Message Authentication Code (cryptography)
MEP	Multipurpose Expansion Port
NMS	Network Management System
NN	Neighbourhood Network
NNAP	Neighbourhood Network Access Points
OBIS	Object Identification System
OSGP	Open Smart Grid Protocol [a)]
OSI	Open System Information
PID	Program Identifier
PLC	Power Line Carrier
PSK	Pre-Shared Key
SAP	Service Access Point
SDU	Service Data Unit
SICB	Service Interface Control Block
UNID	Unique Node Identifier
xDLMS	Extended DLMS

[a)] OSGP is a business model that defines one method for how smart metering should be implemented, and comprises a variety of items and parameters including security design and implementation, data formats, data model, device functionalities.

4 Targeted communication environments

The DLMS/COSEM communication profile over IEC 14908 series networks is intended for remote data exchange on Neighbourhood Networks (NN) between Neighbourhood Network Access Points (NNAP) and Local Network Access Points (LNAPs) or metering devices, using BPSK narrowband PLC technology over the low voltage electricity distribution network as a communication medium. The functional reference architecture is shown in Figure 1.

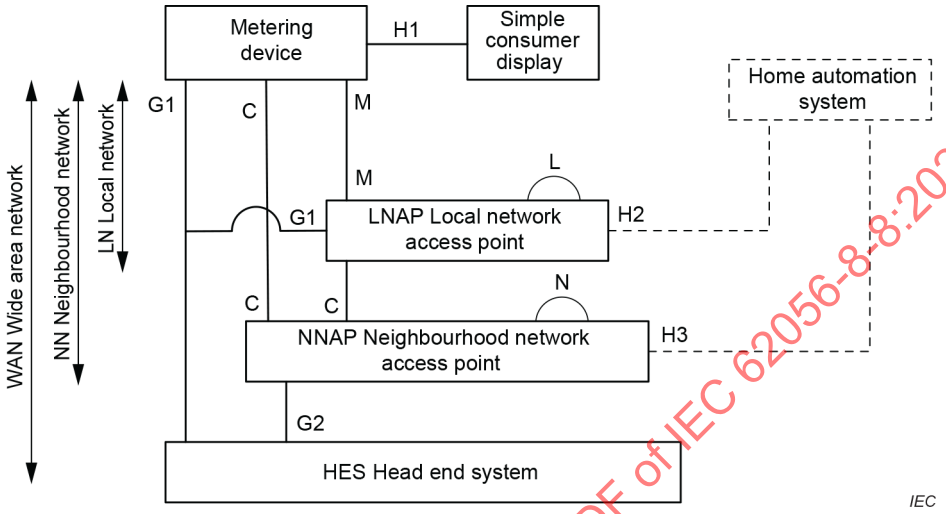

Figure 1 – Functional reference architecture

Metering Devices – typically electricity meters – comprise application functions and communication functions. They may be connected directly to the NNAP via the C interface, or to an LNAP via an M interface, while the LNAP is connected to the NNAP via the C interface. The LNAP function may be co-located with the metering functions.

A NNAP comprises gateway functions and it may comprise concentrator functions. Upstream, it is connected to the Metering Head End System (HES) using suitable communication media and protocols.

Metering Devices and LNAPs may communicate to different NNAPs, but to only one NNAP at a time. From the PLC communication point of view, the NNAP acts as a gateway while metering devices and LNAPs act as endpoints.

NNAPs and similarly LNAPs may communicate with each other, but this is out of the scope of this document, which covers the C interface only.

When the NNAP has concentrator functions, it acts as a DLMS/COSEM client. When the NNAP has gateway functionality only, then the HES plays the role of a DLMS/COSEM client. The metering devices or the LNAPs play the role of DLMS/COSEM servers. In this specification, the NNAP is considered as playing the role of a DLMS/COSEM client, meaning having concentrator functions, therefore, it is also named Data Concentrator (DC).

5 Use of the communication layers for this profile

5.1 Information related to the use of the standard specifying the lower layers

The DLMS/COSEM communication profile over ISO/IEC14908 series networks uses the services of the lower layers specified in ISO/IEC 14908-1:2012 and in ISO/IEC 14908-3:2012.

5.2 The structure of the communication profile

The structure of the DLMS/COSEM communication profile over ISO/IEC 14908 series networks is shown in Figure 2.

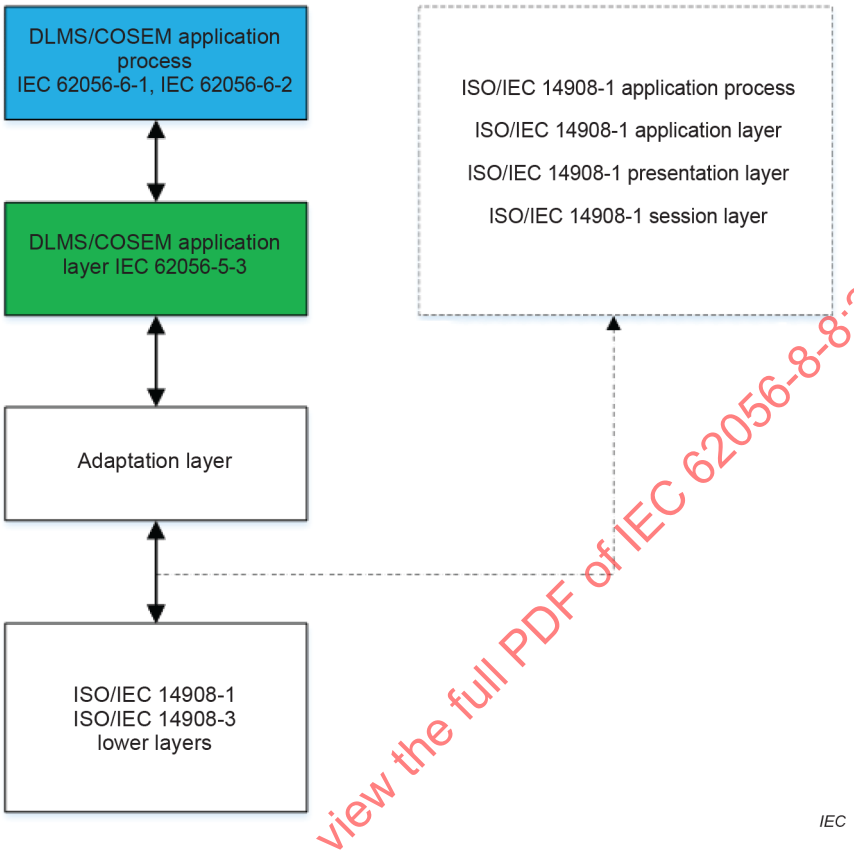

Figure 2 – Structure of the communication profile

The parts shown with solid lines constitute a pure DLMS/COSEM communication profile for ISO/IEC 14908 series networks, using the COSEM data model as specified in IEC 62056-6-1:2017 and IEC 62056-6-2:2017 together with the DLMS/COSEM Application layer as defined in IEC 62056-5-3:2017 and with the lower layers specified in the ISO/IEC 14908 series of standards.

The parts shown with dashed lines are not in the scope of the present document, but are from the ISO/IEC 14908 series of standards.

In the remaining part of this document, all the messages use the big-endian format.

5.3 Lower layers and their use

5.3.1 Overview

The lower layer protocols are specified in ISO/IEC 14908-1:2012 and ISO/IEC 14908-3:2012. They comprise the Physical, the Data link, Network, and Transport layers according to the OSI model.

5.3.2 Physical layer

ISO/IEC 14908-1:2012 allows several types of physical layers. The physical layer used in this standard is for narrowband PLC using BPSK modulation in the CENELEC Frequency Band A. It provides the interface between the equipment and the physical transmission medium, which is the electricity distribution network. It receives and transmits DLPDUs between neighbour nodes by using the PLC technology as defined in ISO/IEC 14908-3:2012, using BPSK modulation in the CENELEC Frequency Band A.

5.3.3 Data link layer

5.3.3.1 Overview

The Data Link layer consists of two parts: the MAC sublayer and the Link layer, as specified in Clauses 6 and 7 of ISO/IEC 14908-1:2012.

5.3.3.2 MAC sublayer

The role of the MAC sublayer is to interface the physical layer and to provide all the necessary mechanisms for the management of the medium in order to liberate the link layer from tasks related to the nature of the medium. This includes predictive p-persistent CSMA algorithm, idle channel detection, the channel backlog estimation and priority slot management. See ISO/IEC 14908-1:2012, 6.1, 6.5 and ISO/IEC 14908-1:2012, Annex A.2.

5.3.3.3 Link layer

The Link layer performs error detection with the means of a CRC. There is no error recovery or acknowledgment available. Corrupted frames are simply discarded. See ISO/IEC 14908-1:2012 Clause 7.

5.3.4 Network layer

The Network layer performs repeating functionality when the network topology is a single channel network topology like bus, token ring, or star topology.

Routing functionality is used when the network topology is a tree or simple topology. A network configuration must be set up first. Addresses used are those configured inside the routers and not those in the PDU exchanged. See ISO/IEC 14908-1:2012, Clause 8.

In the scope of this communication profile the Network layer performs address recognition as specified in ISO/IEC 14908-1:2012, 8.6. It passes inbound packets to the Transport Layer only if the packet is addressed to this node's logical or physical node identifier. Routing is performed by the repeating entity of the Adaptation layer (see Annex E) instead of the Network layer.

This communication profile provides means for application-layer repeating. The layer 3 routing may be disabled if it cannot route as effectively as the repeating algorithm.

5.3.5 Transport layer

5.3.5.1 Overview

The Transport layer consists of the transaction control sublayer and the transport layer itself.

5.3.5.2 Transaction control sublayer

The role of the Transaction control sublayer is to ensure an end-to-end data delivery by providing duplicated packet detection by the receiver, owing to the nature of the network and the possibility for loss of acknowledgment. A transaction consists of a request and a response. A response may contain the requested data, or simply an acknowledgement, or may be absent, at the sender's option. In some situations (like broadcast), a response is not expected. Several mechanisms are available inside the Transaction control sublayer for facilitating detections: identification of the transaction in progress, limitation of the number of concurrent transactions, etc. See ISO/IEC 14908-1:2012, 9.1.

5.3.5.3 Transport layer

The Transport layer controls the end-to-end data transport by verifying that the correct response is received according to the request sent and if not, performs the appropriate retries. See ISO/IEC 14908-1:2012, Clause 10.

5.4 Service mapping and Adaptation layer

5.4.1 Overview

The ISO/IEC 14908-1:2012 protocol stack has, at the top of the Transport layer, the Session, Presentation and Application layers.

In the DLMS/COSEM over ISO/IEC 14908 networks communication profile, an Adaptation layer is located immediately above the Transport layer. The role of the Adaptation layer is to provide the necessary elements for correctly binding the DLMS/COSEM Application layer and the COSEM object model with the lower layers specified in ISO/IEC 14908-1:2012. It also allows the selection of the DLMS/COSEM Application layer and Application process as defined in IEC 62056-5-3:2017 and IEC 62056-6-2:2017.

5.4.2 Adaptation layer

5.4.2.1 General

The Adaptation layer allows the binding of the lower layers with the DLMS/COSEM Application layer.

In addition, it is in charge of the discovery, registration and commissioning processes; see Annex F.

Concerning DLMS/COSEM application layer message exchange, the roles of the Adaptation layer are:

- to provide and manage the path the message shall take for reaching the destination, see 5.4.2.6.2.2.2;
- to redirect the messages to the DLMS/COSEM Application layer depending on the value of the control field during the message reception, see 5.4.2.3.1;
- to provide the required addressing parameters for the client application process and the logical device, see 5.4.2.3.2 items 3 and 4;
- to provide the parameters allowing the correct binding of the higher layer protocol with the Transport layer, see 5.4.2.5;
- to transport the DLMS/COSEM APDUs using the ADP_Data services;
- Security services providing authentication, encryption, and replay prevention are also specified. The main purpose of the Adaptation layer security services is to protect the discovery, registration and commissioning messages. For using Adaptation layer security services to protect the ADP-Data services, refer to 7.4.2.

NOTE These services do not rely on the ISO/IEC 14908-1:2012 session layer security service that provides authentication only.

The Adaptation layer is basically a connectionless protocol. However, a security CryptoContext needs to be established first (see Clause D.7). This CryptoContext establishment is performed during commissioning; see Clause F.5. The CryptoContext may be renewed during the lifetime of the device for example for key updating purposes.

5.4.2.2 Adaptation layer structure

Figure 3 shows the Adaptation layer structure. Although the structure of the Adaptation layer of the NNAP and the LNAP is the same, the content of its elements differs.

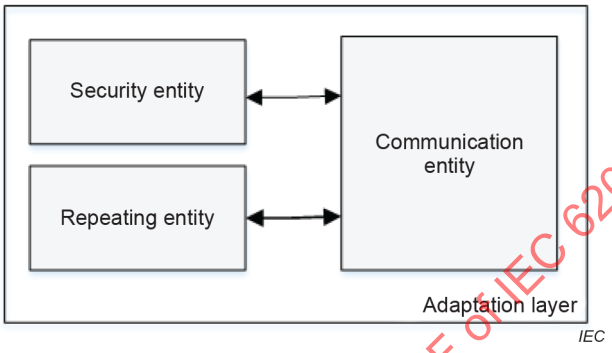

Figure 3 – Adaptation layer structure

The Adaptation layer contains 3 entities:

- a communication entity which is in charge – depending on the configuration of the Adaptation layer – of constructing the PDU for the lower layers during data transmission;
- a security entity which allows the management of the AdpPDU protection, when the Adaptation layer security is activated;
- a repeating entity which allows the indication to the AdpPDU of the path the AdpPDU shall take for reaching the targeted LNAP in the best way.

The communication entity is the same at the NNAP and the LNAP side. It forms, from the service parameters received from the Application layer, the Adaptation layer header during data transmission. During data reception, it processes the Adaptation layer header and presents to the Application layer with the indication primitive, the data received from the other node.

At the NNAP side, a configuration parameter determines if the Adaptation layer security is in force or not. When the Adaptation layer security is in force, a Key Management System (KMS) holds all the symmetric ciphering keys related to all the configured LNAPs inside the network, together with their node identification. It is the scope of the security entity to perform an authenticated encryption of outgoing AdpPDUs ("ciphering") and authenticated decryption and replay prevention of incoming AdpPDUs ("deciphering"). See Annex D for more details.

At the LNAP side, the security entity performs the same functionality as at the NNAP side. The KMS is limited to the Adaptation layer symmetric ciphering key of the related device. The value of the control field received from the NNAP determines if the Adaptation layer security is in force or not.

The repeating entity instructs the message from the NNAP to the LNAP about the best path for reaching the targeted LNAP. The proxy parameters are available in the frames sent by the NNAP only. For more information, refer to Annex E.

5.4.2.3 Adaptation layer PDU structure

5.4.2.3.1 Overview

The Adaptation layer has two kinds of AdpPDUs. An unprotected AdpPDU and a protected AdpPDU, both identified by the Control field as specified in Table 1.

Table 1 shows the possible values the Control field may have and their meaning in the scope of this document.

Table 1 – Control field

Value	Description
0x20	Unprotected AdpPDU
0x21	Protected AdpPDU

The Control field acts as a protocol selector:

- when set to value 0x20, it indicates that the PDU is a DLMS/COSEM Application layer PDU (APDU) with Adaptation layer security not in force;
- when set to value 0x21, it indicates that the PDU is a DLMS/COSEM APDU with Adaptation layer security in force;
- when the value is not 0x20 or 0x21, then the APDU is not a DLMS/COSEM APDU. Refer to ISO/IEC 14908-1:2012, 12.4 in such a case. All the remaining part of this clause is related to DLMS/COSEM over ISO/IEC14908 communication profile.

5.4.2.3.2 NNAP unprotected Adaptation layer PDU structure

The structure of the NNAP unprotected AdpPDU is shown in Figure 4. The AdpPDU contains the header and the payload, and in the downlink parameters, the Proxy parameters are appended:

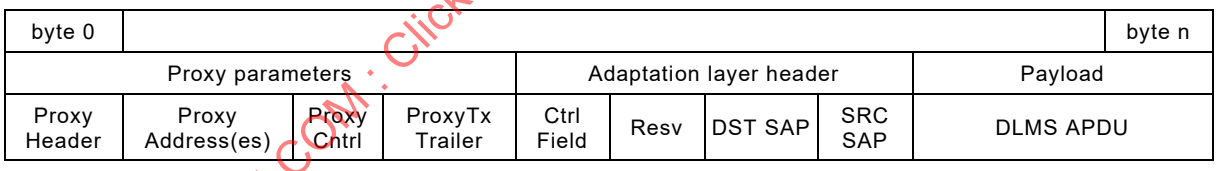

Figure 4 – Unprotected Adaptation layer PDU structure

The proxy parameters contain addressing information required for delivering the message via a series of 1 to 8 repeaters. See Clause E.4.

The NNAP Adaptation layer header consists of the following fields:

a) The Ctrl field has a fixed value set to 0x20;

b) 'Resv' field is of one byte and is reserved for future. It shall be set to 0;

c) 'DST SAP' holds the destination address of the DLMS/COSEM APDU, meaning the logical device SAP when the DLMS/COSEM APDU is from the NNAP to the LNAP and the client SAP when the DLMS/COSEM APDU is from the LNAP to the NNAP. The length of the destination SAP field is of one byte. See Table 2;

d) 'SRC SAP' holds the source address of the DLMS/COSEM PDU meaning the client SAP when the DLMS/COSEM PDU is from the NNAP to the LNAP, and the logical device SAP when the DLMS/COSEM PDU is from the LNAP to the NNAP. The length of the source SAP field is of one byte. See Table 2.

The payload is the DLMS/COSEM APDU. It is the remaining part of the AdpPDU that follows the SRC SAP field, up to the end of the message.

5.4.2.3.3 NNAP protected Adaptation layer PDU structure

When the Adaptation layer security is in force, the structure of the NNAP protected AdpPDU is as shown in Figure 5.

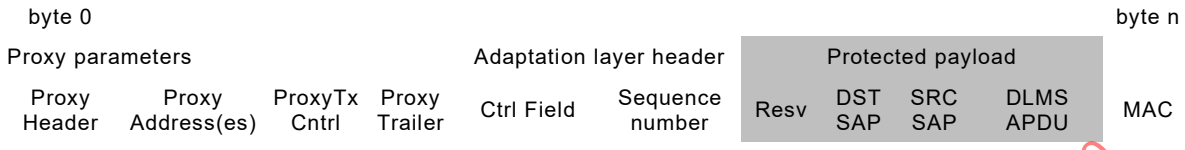

Figure 5 – Protected Adaptation layer PDU structure

where:

- the Ctrl field has a fixed value set to 0x21;
- the Sequence number field is of 4 bytes. It is an increasing value referencing the message. It is reset to 1 every time a new encryption key is derived. It is used for the nonce construction (see Table D.9, and Table D.12) and for the protection against replay attacks; see D.8.4.5;
- the Protected payload, is the encrypted result of the Resv field, the destination SAP, the source SAP and the DLMS/COSEM APDU (see Figure 4), concatenated in this order. These fields are encrypted and authenticated using the algorithms as described in Clause D.4. For more information see Annex D;
- the message authentication code (MAC). The length of the MAC is 8 bytes. The MAC is related to the Protected payload. For more information, refer to Annex D.

5.4.2.3.4 LNAP Adaptation layer structure

When the frame is from the LNAP to the NNAP, the AdpPDU structure is the same as the one specified in 5.4.2.3.2 and 5.4.2.3.3, except that the Proxy parameters are not present.

5.4.2.4 Adaptation layer maximum data length

The maximum length of the AdpPDU is limited to 228 bytes.

When protection is not in force, this length includes:

- the Proxy parameters, which may be up to 48 bytes (a maximum of 8 repeaters is allowed);
- the Adaptation layer header;
- the payload i.e. the DLMS/COSEM APDU.

When the Adaptation layer protection is in force, it includes also the lengths of the Sequence number parameter and the MAC.

5.4.2.5 Adaptation layer services

5.4.2.5.1 Service mapping

The role of the Adaptation layer is to redirect the APDU to the DLMS/COSEM Application layer depending on the value of the control field (see Table 1), and to provide the necessary procedures and services for binding the lower layers to the Application layer. Figure 6 presents the services offered by the Adaptation layer.

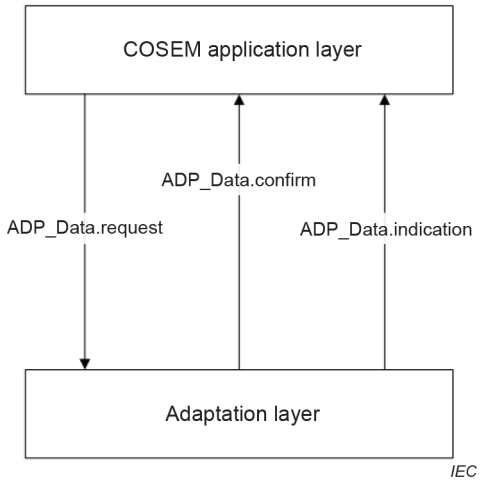

Figure 6 – Adaptation layer services

5.4.2.5.2 Service specification

5.4.2.5.2.1 General

As the protocol of the Adaptation layer is connectionless protocol, it provides only ADP_Data services: ADP_Data.request, ADP_Data.confirm and ADP_Data.indication.

5.4.2.5.2.2 ADP_Data.request

Function

The ADP_Data.request primitive is invoked by the DLMS/COSEM Application layer to the Adaptation layer to request sending an Adaptation layer PDU to the peer Adaptation layer entity using the transmission procedures.

Semantics

The semantic of the primitive is as follows:

```
ADP_Data.request
(
        service_type,
        device_address,
        destination_SAP,
        source_SAP,
        Data
)
```

where:

The service_type parameter specifies which TPDU type should be used for transporting the Data. The service_type parameter is mapped from the Invoke-Id-And-Priority service class field of the xDLMS services or the response-allowed parameter of the InitiateRequest APDU:

- when the service_class field of the Invoke-Id-And-Priority parameter is of type confirmed or the response-allowed parameter or the InitiateRequest APDU is TRUE, then the service_type should be of type confirm;
- when the service_class field of the Invoke-Id-And-Priority parameter is of type unconfirmed or the response-allowed parameter of the InitiateRequest APDU is FALSE, then the service_type should be of type unconfirm.

Table 3 specifies the mapping of the service-class field of the Invoke-Id-And-Priority and response-allowed parameters into the service_type parameter.

The device_address parameter contains the destination ISO/IEC 14908 address of the device the data has to be delivered to. It is made up of the subnet_ID and the node_ID. The domain_ID is a configuration parameter populated in the NNAP during commissioning.

The destination_SAP parameter specifies the address of the remote SAP involved in the Data transmission.

The source_SAP parameter specifies the address of the SAP where the service is requested from.

The Data parameter specifies the PDU to be transferred using the Adaptation layer services.

Use

This primitive is used by the DLMS/COSEM Application layer when it has Data to send to the destination Application layer.

When the Adaptation layer receives this primitive, it will construct the AdpPDU to make available for the lower layers by inserting the Control field, the reserved field, the destination_SAP and the source_SAP at the beginning as specified in Figure 4, and if the protection is in force, then applies the protection (see Figure 5).

If the sender is the NNAP, then the proxy parameters are also inserted. See Annex E for the construction of the proxy parameters.

Once this process is done, the Adaptation layer invokes the lower layer data transmission primitive.

5.4.2.5.2.3 ADP_Data.confirm

Function

The ADP_Data.confirm primitive issues a local confirmation upon the completion of the Data transmission to the lower layer, at the Adaptation layer level. The scope of this primitive is only local. It notifies the Application layer – having invoked an ADP_Data.request primitive – of the result of the processing of the related request. This result may be SUCCESS or FAILED.

Semantics

The semantic of the primitive is as follows:

```
ADP_Data.confirm
(
        device_address,
        destination_SAP,
        source_SAP,
        status
)
```

The device_address parameter contains the destination ISO/IEC 14908 address of the device the data is delivered to. The value of this parameter is the same as the one provided in the corresponding ADP_Data.request primitive.

The destination_SAP parameter specifies the address of the remote SAP involved in the data transmission. The value of this parameter is the same as the one provided in the corresponding ADP_Data.request primitive.

The source_SAP parameter specifies the address of the SAP where the service is requested from. The value of this parameter is the same as the one provided in the corresponding ADP_Data.request primitive.

The status parameter is used for providing a local result (SUCCESS or FAILED) of the data transmission related to the ADP_Data.request previously issued.

Use

This primitive is locally generated by the Adaptation layer to the local Application layer entity as a response to an ADP_Data.request primitive to inform the originator of the request about the result of the data transmission procedures.

5.4.2.5.2.4 ADP_Data.indication

Function

The ADP_Data.indication primitive is passed from the Adaptation layer entity to the Application layer entity for notifying it about the arrival of an AdpPDU from the peer station or an error from the Transport layer. This primitive defines the transfer of Data from the Adaptation layer entity to the Application layer entity.

Semantics

The semantic of the primitive is as follows:

```
ADP_Data.indication
(
        service_type,
        source_address,
        destination_SAP,
        source_SAP,
        status,
        Data
)
```

The service_type is a parameter indicating the type of service: whether it is confirm or unconfirm service type.

The source_address parameter is the ISO/IEC 14908 physical address of remote device the Data is originated from. The source_address parameter is provided to the Adaptation layer by the Transport layer. See ISO/IEC 14908-1:2012, 8.3, Rcv_Packet(address_pair, pduType, PDU, priority) primitive.

The destination_SAP parameter specifies the address of the logical device SAP when the DLMS/COSEM APDU is from the NNAP to the LNAP, and the client_ID when the DLMS/COSEM APDU is from the LNAP to the NNAP. The destination_SAP parameter is part of the Adaptation layer header as provided by the remote device. See 5.4.2.3.2.

The source_SAP parameter specifies the address of the SAP of the application process that sent the DLMS/COSEM APDU. The source_SAP parameter is part of the Adaptation layer header as provided by the remote device. See 5.4.2.3.2.

The status parameter informs the Application layer about the arrival of Data. In case of failure, the status parameter provides an error code and the Data parameter is simply empty. See Clause D.12.

The Data parameter carries the DLMS/COSEM APDU received using the Adaptation layer service. See 5.4.2.3.2.

Use

The ADP_Data.indication primitive is passed from the Adaptation layer entity to the Application layer entity for informing the Application layer entity about the arrival of Data from a remote Adaptation layer user entity. NNAP requests are notified to the LNAP by a ADP_Data.indication primitive at the LNAP Application layer level. In turn, LNAP responses to NNAP requests are notified to the NNAP by an indication at the NNAP Application layer level.

5.4.2.6 Protocol specification

5.4.2.6.1 Overview

The data exchange is based on *client-server* paradigm. The client sends a request to the server and then – if the request is of type confirmed – the server responds to this request as shown in Figure 7.

Data transmission is initiated upon the initiative of the DLMS/COSEM client Application layer when this entity has a request to send. When the DLMS/COSEM client Application layer invokes the ADP_Data.request, it provides with the request the physical device destination address, the client SAP and the server SAP. The Adaptation layer processes this request and invokes the Transport layer services for transmission.

At the LNAP side, the server Adaptation layer receives the data from its Transport layer and processes it in order to provide the relevant ADP_Data.indication to its Application layer. This indication carries the request as sent by the client to which the server Application layer has to respond, depending on the indication service type. The response, if any, is provided by the Application layer with the means of an ADP_Data.request primitive to its Adaptation layer, which has to process it and make it available to the Transport layer for transmission.

At the NNAP side, the client Adaptation layer receives the data from its Transport layer and processes it in order to provide the relevant ADP_Data.indication to its Application layer. This indication carries the response as sent by the server to its request.

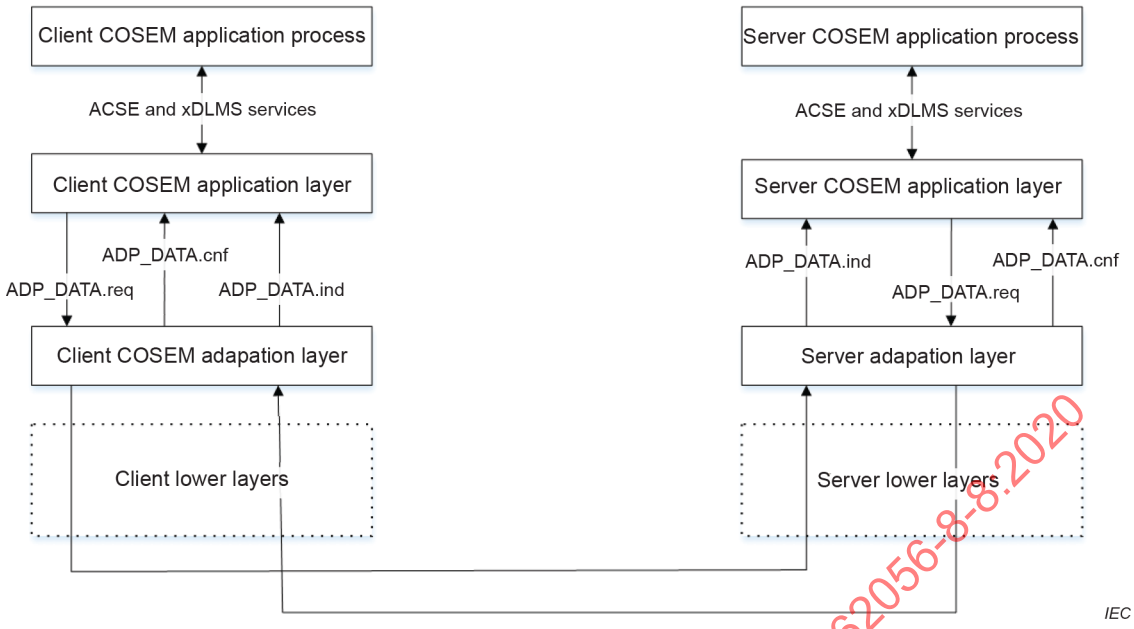

Figure 7 – Client and server ADP_DATA services

5.4.2.6.2 Request management

5.4.2.6.2.1 General

Any confirmed request received at the Adaptation layer level needs to be responded to (meaning that an ADP_Data.indication has to be invoked by the Adaptation layer) before the next ADP_Data.request can be invoked by the Application layer. Cancellation is provided by the Transport layer when its transaction fails. See 5.3.5.2. Any new request received from the Application layer, before the response to the previous one is received, is simply discarded.

5.4.2.6.2.2 Sending data – NNAP side

5.4.2.6.2.2.1 Adaptation layer header

The NNAP Adaptation layer maintains, as specified in 5.4.2.2, a configuration parameter which indicates if protection is in force or not. When the protection is in force, it concerns all the nodes available in the subnetwork. In this case, a KMS (Key Management System) is available in the NNAP. It holds all the security keys related to each node of the subnetwork.

When the NNAP Adaptation layer is requested to transmit data to a given LNAP, it first inserts the relevant Control field depending on whether the protection is force or not.

After that, it inserts the reserved field, the destination SAP, the source SAP followed by the Application layer APDU as provided by the Application layer.

If the protection is not in force, then the message that is built is passed to the repeating entity for the inclusion of the proxy parameters. The AdpPDU as constructed is made available for the Transport layer, using the appropriate primitive.

When the protection is in force, the Adaptation layer constructs the AdpPDU by inserting the relevant Control field, and then the reserved field, the destination SAP, the source SAP, followed by the APDU as provided by the Application layer. After this, the Adaptation layer requests the protection of the message to be built by the security entity. This protection consists of the encryption then the MAC calculation of the part of the message, starting with the reserved field followed by the destination and the source SAPs, and the DLMS/COSEM APDU provided by the DLMS/COSEM Application layer, using the symmetric key shared by the NNAP and the relevant LNAP. Once this operation is performed, the Adaptation layer security entity:

- inserts the four-bytes request sequence number immediately after the Control field;
- inserts the encrypted message that is constructed;
- and then, adds the calculated MAC after the protected payload, see Figure 5.

The message that is built is passed to the repeating entity for the inclusion of the proxy parameters. The AdpPDU as constructed is made available to the Transport layer, using the appropriate primitive.

5.4.2.6.2.2.2 Proxy parameters

The exchanges are based on the master/slave principle where the slaves (the LNAPs) are only allowed to transmit upon the invitation of the NNAP. A typical network is controlled by the NNAP, which also plays the role of the network topology manager. All LNAPs have repeating capability. To extend the communication range, the Repeating entity of the NNAP Adaptation layer may instruct the LNAPs inside the communication network about the best path the Application message should take to reach the destination. This path is specified in the Proxy parameters carried by the AdpPDU. The message is then transferred from hop to hop, up to the final destination. For more details, refer to Annex E.

Internally, the Repeating entity of NNAP Adaptation layer manages a repeating table that allows it to have knowledge of the order of the repeater LNAPs needed, hop by hop, for reaching each destination (see E.2.3) available in the subnetwork. For a given destination device address, the Repeating entity of the NNAP Adaptation layer extracts LNAP addresses from the repeating table for repeaters, and then builds the Proxy parameters field of the AdpPDU. The Proxy parameters are as specified in Figure 4 and Figure 5, meaning that the Proxy Header, the Proxy Addresses, the Proxy Control and Proxy Trailer fields are inserted at the beginning of the frame as specified in Clause E.3.

5.4.2.6.2.2.3 Lower layer service invocation

Once the operations specified in 5.4.2.6.2.2.2 are completed, the Adaptation layer makes the message available to the Transport layer with the appropriate service_type in the service primitive; see ISO/IEC 14908-1:2012, 10.4:

- when the service_type parameter is of type confirm, then the TPDU type to use should be request/response;
- when the service_type parameter is of type unconfirm, then the TPDU type to use should be Unackd-Rpt.

5.4.2.6.2.2.4 Transport layer confirmation

After making available the message to the Transport layer for transmission, the Adaptation layer then builds and sends a successful ADP_Data.confirm primitive to the Application layer whenever the AdpPDU has been successfully provided to the Transport layer. Otherwise, the Transport layer communicates a failure to the Adaptation layer.

5.4.2.6.2.3 Sending data – Server side

5.4.2.6.2.3.1 Adaptation layer header

The LNAP always sends a response to a confirmed request coming from the NNAP after the LNAP Application layer issues an ADP_Data.request primitive with a confirm service type (a LNAP can never send an unconfirm request). Applying the protection is specified by the control field:

a) if the Control field of the request received from the NNAP is equal to Protected AdpPDU (value 0x21, see Table 1), then the response to be sent by the LNAP shall be a protected payload;

b) if the Control field of the request received from the NNAP is an Unprotected AdpPDU (value 0x20, see Table 1), then the response to be sent by the LNAP shall be a non-protected payload.

When the LNAP Adaptation layer is requested to transmit data to the NNAP, it simply copies the Control field received from the NNAP into the response, adds the reserved field, and then fills the destination and source SAP fields, followed by the Application layer PDU.

If the protection is not in force, the Adaptation layer makes the message available to the Transport layer, because the Proxy parameters are not needed for AdpPDUs from the LNAP to the NNAP. The source and destination SAPs are provided in the ADP_Data.request originated by the Application layer.

When the security is in force, the LNAP Adaptation layer communication entity performs the sequence described above for the non–protected payload, then passes the data to the security entity, which performs the protection by making use of the security material available. Each server stores, at the Adaptation layer level, all the security parameters needed for the Adaptation layer payload protection. These security parameters are:

- the unicast encryption key (UCK) and the broadcast encryption key (BCK), which are obtained via key derivation, see Clause D.6;
- the unicast request and receive sequence numbers and the receive broadcast sequence number, which are of 4 bytes each. See D.8.3.3 and D.8.4.3;
- the meter unique identifier and the domain-ID for the nonce calculation.

The protection consists of the encryption and the MAC calculation for the part of the message, made up of the reserved field, the destination and source SAPs and the DLMS/COSEM APDU.

The Adaptation layer communication entity then concatenates to the Control field the transmit Sequence number, the protected payload as prepared by the security entity, and then the MAC. Then it makes the message available to the Transport layer.

5.4.2.6.2.3.2 Proxy parameters

There are no proxy parameters when the response message is transmitted from the LNAP to the NNAP.

5.4.2.6.2.3.3 Lower layer service invocation

At the LNAP side, the Adaptation layer communication entity does not process the service_type field. An ADP_Data.request sent by the LNAP Adaptation layer is always of confirm type.

5.4.2.6.2.3.4 Transport layer confirmation

After making available the data to the Transport layer for transmission, the Adaptation layer communication entity builds and sends a successful ADP_Data.confirm to the Application layer whenever the AdpPDU is successfully provided to the lower layer. Otherwise, the Transport layer communicates a failure to the Adaptation layer communication entity. See ISO/IEC 14908-1:2012, 11.3.

5.4.2.6.3 Receiving data – NNAP side

5.4.2.6.3.1 Proxy parameters management

The AdpPDU from the LNAP to the NNAP never carries Proxy parameters.

5.4.2.6.3.2 Adaptation layer header

During reception, the Adaptation layer receiving entity begins with the analysis of the Control field:

a) if the Control field is equal to Unprotected AdpPDU (value = 0x20), then the communication entity of the Adaptation layer extracts the destination and source SAP and the APDU from the AdpPDU received, and then builds the ADP_Data.indication primitive for the Application layer;

b) if the Control field is equal to Protected AdpPDU (value = 0x21), then the Adaptation layer communication entity passes the data to the security entity, which verifies and removes the protection. The protection verification consists of:

- checking the received sequence number carried by the message, which has to be greater than the last sequence number stored in the NNAP security entity for the related LNAP;
- the deciphering and the verification of the MAC using the security material available in the configuration parameters for the related LNAP.

When the protection verification is successful, the Adaptation layer communication entity uses the destination and source SAPs and the APDU carried by the frame, to invoke the ADP_Data.indication primitive in order to notify the Application layer of the availability of the data received.

When the protection verification of the request message fails, then the data is simply discarded and the Adaptation layer forms and sends an error response to the Application layer with the indication primitive. In such a case, the Data field is simply void. See Clause D.12.

5.4.2.6.4 Receiving data – LNAP side

5.4.2.6.4.1 Proxy parameters management

At each hop, once a given LNAP Repeater notices that it is not the final destination of the frame, it consumes its own address in the frame and redirects the remaining part of the frame to the next hop, and so on, until the frame reaches the destination. For more information, see Annex E.

5.4.2.6.4.2 Adaptation layer header

The protocol for receiving is the same at the NNAP and the LNAP Adaptation layer.

5.4.2.7 Segmentation and reassembly

Segmentation is not supported. See also 7.5.

5.5 Registration and connection management

The registration and connection management is specified in Annex F.

6 Identification and addressing schemes

During the registration process, an LNAP obtains from the NNAP a physical device address which is made up of the domain_ID, the subnet_ID and the node_ID. In the scope of this specification, the size of the domain_ID shall be of 3 bytes. See ISO/IEC 14908-1:2012, Clauses E.1, E.2 and E.6.

The physical device source and destination addresses are transported within every frame using the MAC sublayer services.

At the top of the protocol stack on the NNAP side, the client Application Process uses the client SAP for its addressing. Several client Application Processes may be located inside the same physical entity. Each of them is identified by its SAP.

At the top of the protocol stack at the LNAP side, several logical devices may be located inside the same physical device. Each logical device can be reached at its SAP.

An Application Association established between a specific client and a specific logical device is identified by the {(Client Physical device address, Client ID), (Server Physical device address, Server logical device ID)} doublet.

Table 2 provides the SAPs for the client and the server.

Table 2 – Client and Server SAP

Client SAPs	
No-station	0x00
Client Management Process	0x01
Public Client	0x10
Open for client SAP assignment	0x02… 0x0F and 0x11 to FF
Server SAPs	
No station	0x00
Management Logical Device	0x01
Reserved for future use	0x02… 0x0F
Open for server SAP assignment	0x10…0x7E
All-station (Broadcast)	0xFF

7 Specific considerations for the application layer services

7.1 Overview

The present clause defines the specific constraints for using the DLMS/COSEM application layer in this communication profile.

7.2 Application Association establishment and release: ACSE services

7.2.1 Application association establishment

To be able to establish an AA between the LNAP and the NNAP managing the subnetwork, the LNAP has to discovered and configured first.

As specified in IEC 62056-5-3:2017, AAs may be explicitly established or preestablished. To explicitly establish AAs, the client uses the ACSE services. No specific constraints apply.

Both explicitly established and preestablished AAs may be confirmed or unconfirmed.

Table 3 shows the rules for explicitly establishing confirmed and unconfirmed AAs, the type of data transfer services available in such AAs and the AdpPDU types used for carrying APDUs. In this table, grey areas represent cases which are out of the normal operating conditions: either not allowed or have no useful purpose.

Table 3 – Application associations and data exchange in the DLMS/COSEM over ISO/IEC14908 profile

Application association establishment				Data exchange	
Protocol connection parameters	**COSEM-OPEN service class**	**Use**	**Type of established AA**	**Service class**	**ADP-Data service type**
Phy device address: {domain ID, subnet_ID, node_ID}	Confirmed	AARQ/AARE using ADP-Data services	Confirmed	Confirmed data exchange allowed	ADP-Data confirm
				Unconfirmed data exchange allowed	ADP-Data unconfirm
	Unconfirmed		Unconfirmed	Confirmed data exchange not allowed	-
				Unconfirmed data exchange allowed	ADP-Data unconfirm

Data exchange using confirmed and/or unconfirmed services may take place in a confirmed AA.

Data exchange using a confirmed service cannot take place in an unconfirmed AA.

7.2.2 Application association release

The layer (N-1) of the Application layer, meaning the Adaptation layer in this profile, has a connectionless oriented protocol. Therefore, the AA can only be closed with the means of the ACSE A-RELEASE services – by invoking the COSEM-RELEASE.request primitive.

7.2.3 COSEM-OPEN and COSEM-RELEASE service parameters

The service parameters of the COSEM-OPEN service shall be used as described below:

The Protocol_Connection_Parameters parameters shall be:

Protocol (Profile) Identifier	ISO/IEC 14908
Server_subnet_ID	Subnetwork to which the device belongs [1)]
Server_node_ID	Node identifier assigned to the device during the registration
Server_SAP	COSEM Logical Device Address SAP
Client_subnet_ID,	Subnetwork to which the client belongs [1)]
Client_node_ID	Node Identifier assigned to the client
Client_SAP	COSEM client application process (type) identifier

[1)] See E.3.2.

The User_information parameter service is not used.

The service_class parameter is specified in Table 3.

The Use_RLRQ_RLRE parameter of the COSEM-RELEASE.request primitive shall be set to TRUE. The User_information parameter service is not used.

7.3 xDLMS services

The communication profile specified in this document allows the use of all xDLMS services specified in IEC 62056-5-3:2017 standard within an application context using LN referencing or SN referencing except that – due to the transaction principle of the lower layers – unsolicited services cannot be used.

7.4 Security mechanisms

7.4.1 DLMS/COSEM security

The DLMS/COSEM security mechanisms specified in IEC 62056-5-3:2017 and in IEC 62056-6-2:2017 apply at the application layer and the application process level. They can be used without any restriction.

7.4.2 Adaptation layer security

7.4.2.1 Introduction

Applying adaptation layer security to ADP_Data messages are controlled by the Control field of those messages, see 5.4.2.3.1.

7.4.2.2 Security suite

7.4.2.2.1 Overview

This document supports OSGP-AES-128-PSK security suite as specified in D.2.5.

7.4.2.2.2 OSGP-AES-128-PSK

The OSGP-AES-128-PSK (see IETF RFC 5433) security suite provides a secure, mutually authenticated, two-way communication channel between a NNAP and a LNAP. The secure channel provides data confidentiality, integrity, authentication and replay protection.

7.4.3 Lower layers security

In addition to the DLMS/COSEM security and the Adaptation layer security, the lower layers may present security. These security features are out of the scope of this document.

7.5 Transferring long application messages

Transferring long application messages is achieved with the means of the DLMS/COSEM General Block Transfer mechanism that supports confirmed transfer with streaming and block recovery as well as unconfirmed transfer with streaming. This mechanism may convey any large message.

xDLMS service specific block transfer may also be used in the scope of point to point communication.

7.6 Media access, bandwidth and timing considerations

At the Application process and Application layer levels, there are no specific considerations. All the timing and bandwidth constraints are managed at the Transport and Physical layer level.

7.7 Other considerations

Void.

8 Communication configuration and management

The communication configuration parameters are managed with the means of the DLMS/COSEM IEC 14908 interface classes as specified in Annex B. Annex F specifies the Registration and Connection management.

9 The COSEM Application Process

All the features, as defined in the IEC 62056-6-1:2017, IEC 62056-6-2:2017 and IEC 62056-5-3:2017, using the request/response paradigm may be used without any restriction.

See also Annex C that defines a migration path from applications using Utility Tables to full COSEM based applications.

10 Additional considerations for the use of this profile

Void.

Annex A
(informative)

Examples

Void.

Annex B
(normative)

Interface classes

B.1 Interface classes

B.1.1 ISO/IEC 14908 Identification

Instances of the ISO/IEC 14908 Identification interface class allow the identification of the device and the network to which the device is connected to.

ISO/IEC14908 Identification		0...n	class_id = 130, version = 0			
Attributes		***Data type***	***Min.***	***Max.***	***Def.***	**Short name**
1. logical_name	(static)	octet-string				x
2. node_Id	(static)	unsigned	1	127		x + 0x08
3. subnet_ID	(static)	unsigned	1	255		x + 0x10
4. domain_ID	(static)	octet-string				x + 0x18
5. program_ID	(static)	octet-string				x + 0x20
6. unique_node_ID	(static)	octet-string				x + 0x28
Specific methods		***m/o***				

Attribute description	
logical_name	Identifies the "ISO/IEC 14908 Identification" object instance. See Clause B.2.
node_ID	Identification of the node.
subnet_ID	Identification of the subnet to which the device is connected to.
domain_ID	Identification of the network. It shall be 3 bytes.
program_ID	Uniquely identifies the functionalities and version of the firmware running on the device. It shall be 8 bytes.
unique_node_ID	Identification of the physical connection module connected to the ISO/IEC 14908-1:2012 network. Each compliant module is assigned a unique 48-bit identifier called Unique_Node_ID. This ID is unique worldwide and is set at the time of manufacture. The value of this identifier does not change from the time of manufacture.

B.1.2 ISO/IEC 14908 Protocol setup

Instances of the ISO/IEC 14908 Protocol setup interface class allow the configuration of the ISO/IEC 14908 device.

ISO/IEC 14908 Protocol setup		0...n	class_id = 131, version = 0			
Attributes		***Data type***	***Min.***	***Max.***	***Def.***	**Short name**
1. logical_name	(static)	octet-string				x
2. inactivity_timeout	(static)	long-unsigned				x + 0x08
Specific methods		***m/o***				

Attribute description	
logical_name	Identifies the "ISO/IEC 14908 Protocol setup" object instance. See Clause B.2.
inactivity_timeout	Minutes of absence of Adaptation layer messages from the NNAP before the LNAP is considered to be out of communication; see F.4.3.

B.1.3 ISO/IEC 14908 Protocol status

Instances of the ISO/IEC 14908 Protocol status interface class allow having knowledge about the status of the protocol in the ISO/IEC 14908 device.

ISO/IEC 14908 Protocol status		0...n	class_id = 132, version = 0			
Attributes		***Data type***	***Min.***	***Max.***	***Def.***	**Short name**
1. logical_name	(static)	octet-string				x
2. transmission_errors	(dyn)	long-unsigned				x + 0x08
3. transmit_tx_failure	(dyn)	long-unsigned				x + 0x10
4. transmit_tx_retries	(dyn)	long-unsigned				x + 0x18
5. receive_tx_full	(dyn)	long-unsigned				x + 0x20
6. lost_messages	(dyn)	long-unsigned				x + 0x28
7. missed_messages	(dyn)	long-unsigned				x + 0x30
8. layer2_received	(dyn)	long-unsigned				x + 0x38
9. layer3_received	(dyn)	long-unsigned				x + 0x40
10. messages_received	(dyn)	double-long-unsigned				x + 0x48
11. messages_validated	(dyn)	double-long-unsigned				x + 0x50
Specific methods		***m/o***				

Attribute description	
logical_name	Identifies the "ISO/IEC 14908 Protocol status" object instance. See Clause B.2.
transmission_errors	This is a count of the number of transmission errors that have occurred on the network. A transmission error is detected via a CRC error during packet reception or as a packet that is less than 8 bytes long. This could result from a collision, a noisy medium, signal attenuation, etc. See ISO/IEC 14908-1:2012, 13.8.2.
transmit_tx_failure	The number of times that the node failed to receive expected acknowledgments or responses after retrying the configured number of times. See ISO/IEC 14908-1:2012, Clause B.8.

transmit_tx_retries	The number of retries sent by this node. This does not include retries used for messages sent with repeated service. See ISO/IEC 14908-1:2012, Clause B.8.
receive_tx_full	The number of times that an incoming packet was discarded because there was no room in the transaction database. See ISO/IEC 14908-1:2012, Clause B.8.
lost_messages	This is the number of messages that were addressed to the node that were thrown away because there was no application buffer available for the message. See ISO/IEC 14908-1:2012, 13.8.2.
missed_messages	This is the number of messages that were on the network but could not be received because there was no network buffer (packet buffer) available for the message or the network buffer was too small to receive the message. See ISO/IEC 14908-1:2012, 13.8.2.
layer2_received	The number of Layer-2 messages received by this node. Layer-2 messages are those that have correct CRC and can be addressed to any node. See ISO/IEC 14908-1:2012, B.8.
layer3_received	The number of messages transmitted from layer 3 of the protocol processor. These can include network variable updates, explicit messages, acknowledgments, retries, reminders, service pin messages, and any other type of message. See ISO/IEC 14908-1:2012, Clause B.8.
messages_received	Number of messages received, which are delivered to the Adaptation layer.
messages_validated	Number of correct messages received at Adaptation layer level.

B.1.4 ISO/IEC 14908 Diagnostic

Instances of the ISO/IEC 14908 Diagnostic interface class allow to have knowledge about the device status inside the PLC network.

ISO/IEC 14908 Diagnostic		**0...n**	**class_id = 133, version = 0**			
Attributes		***Data type***	***Min.***	***Max.***	***Def.***	**Short name**
1. logical_name	(static)	octet-string				x
2. plc_signal_quality_status	(dyn)	enum				x + 0x08
3. com_module_state	(dyn)	enum				x + 0x10
4. received_message_status	(dyn)	unsigned				x + 0x18
5. no_receive_buffer	(dyn)	long-unsigned				x + 0x20
6. transmit_no_data	(dyn)	long-unsigned				x + 0x28
7. backlog_overflows	(dyn)	long-unsigned				x + 0x30
8. late_ack	(dyn)	long-unsigned				x + 0x38
9. frequency_invalid	(dyn)	long-unsigned				x + 0x40
Specific methods		***m/o***				

Attribute description	
logical_name	Identifies the "ISO/IEC 14908 Diagnostic" object instance. See Clause B.2.
plc_signal_quality_status	Provide the PLC signal quality status: enum (0) No PLC traffic detected, (1) PLC traffic detected, but no traffic addressed to this device, (2) A packet addressed to this device has been received. The signal margin is less than or equal to 9 dB, (3) A packet addressed to this device has been received. The signal margin is between 12 dB or 15 dB, (4) A packet addressed to this device has been received. The signal margin is greater than or equal to 18 dB, (5) Meter is commissioned but its inactivity_timeout as defined in B.1.2. has expired. The last packet addressed to this meter had a signal margin is less than or equal to 9 dB, (6) Meter is commissioned but its inactivity_timeout as defined in B.1.2. has expired. The last packet addressed to this meter had a signal margin at 12 dB or 15 dB, (7) Meter is commissioned but its inactivity_timeout as defined in B.1.2. has expired. The last packet addressed to this meter had a signal margin greater than or equal to 18 dB This parameter is updated every time a packet is received by the device.
com_module_state	Last received state of the device's power line ISO/IEC 14908 communication module: (0, 1) Invalid states, (2) Unconfigured, (3) Applicationless, (4) Configured, (6) Hard-offline, (12) Configured, soft offline, (140) Configured, in bypass mode, (255) No reply when status requested This field is updated on each device power-up, each re-synch request from the NNAP, and each communication module reset. Any value which is not listed above is invalid state. See ISO/IEC 14908-1:2012,13.4, and 13.8.4

received_message_status	Status of the received message: b0: PLC Link Explicit Message Received, b1: reserved, b2: reserved, b3: reserved, b4: reserved, b5 – b7: Transceiver phase 0: No phase inversion present, 1: In Phase normal, 2: Plus 120 degrees inverted, 3: Minus 120 degrees normal, 4: Phase 180 degrees inverted, 5: Plus 120 degrees normal, 6: Minus 120 degrees inverted, 7: N.A
no_receive_buffer	Number of times the ISO/IEC 14908-1:2012 lower layers detected a network inbound packet addressed to it, but has no buffer to store it. Hence the message could not be made available to the Adapation layer.
transmit_no_data	Number of times the Adaptation layer attempted to transmit data to the ISO/IEC 14908-1:2012 stack but no buffer is available. Updated on occurrence. ISO/IEC 14908-1:2012,13.4.
backlog_overflows	Number of times the MAC layer predictive backlog counter overflowed. See ISO/IEC 14908-1:2012, Clause B.8.
late_ack	Number of times an ACK or response was received for a transaction that has been already completed. Typically, this occurs owing to too short transaction timer values or network congestion.
frequency_invalid	Number of times the ISO/IEC 14908-1:2012 stack detects an invalid frequency. Updated on occurrence.

B.2 Relation to OBIS

For setting up and managing data exchange using ISO/IEC 14908 PLC networks at least one instance of each of the following interface classes shall be implemented:

	IC	OBIS identification					
		A	B	C	D	E	F
ISO/IEC 14908 Identification	130, ISO/IEC 14908 Identification	0	*b*	32	0	0	255
ISO/IEC 14908 Protocol setup	131, ISO/IEC 14908 Protocol setup	0	*b*	32	1	0	255
ISO/IEC 14908 Protocol status	132, ISO/IEC 14908 Protocol status	0	*b*	32	2	0	255
ISO/IEC 14908 Diagnostic	133, ISO/IEC 14908 Diagnostic	0	*b*	32	3	0	255

Annex C
(informative)

Implementation guide and migration path

C.1 General

The purpose of this informative Annex is to facilitate implementation of the communication profile specified in this document by recapitulating some basic principles of DLMS/COSEM specified in IEC 62056-5-3:2017 and in IEC 62056-6-2:2017 and showing a migration path from ANSI Utility table based applications to COSEM object-based applications, via the COSEM "Utility tables" objects.

It also specifies a way to minimize the number of exchanges and transferring some long messages – the object_lists – in consideration of the limitations imposed by the lower layers of the communication profile specified in this document. This is achieved by using pre-established Application Associations and defining the list of COSEM objects with their base_names that can be used in actual implementations.

C.2 Client-server model

C.2.1 General

As specified in IEC 62056-5-3:2017, 4.2, DLMS/COSEM is based on the client-server model in which client and server application processes communicate with each other. The clients send service requests to servers and the servers respond to these requests. This takes place within Application Associations, see C.2.4.

In the communication environments targeted by the communication profile specified in this document the client APs are located in the NNAP and server APs are located in LNAPs which may be co-located with the end devices. See also Clause 4, Figure 1.

C.2.2 Clients

A server may be accessed by several clients each using specific contexts and having specific access rights.

IEC 62056-5-3:2017, 4.1.5 specifies three different kinds of contexts:

- the application context that stipulates the referencing style of COSEM object attributes and methods and the use of ciphering;
- the authentication context, that stipulates the method and algorithms to authenticate the client and the server;
- the xDLMS context that stipulates the set of xDLMS services to be used and some other parameters.

The access rights determine the list of COSEM objects visible in a given AA, together with the access rights to their attributes and methods, as well as the required protection of the service requests and responses.

There shall be at least one public client implemented. Its role is revealing the internal structure of the servers. It has a pre-defined Service Access Point; see Table 2.

See also IEC 62056-5-3:2017, 4.1.14.

C.2.3 Servers

As specified in IEC 62056-6-2:2017, 4.7, the server is a physical device that contains one or more logical devices. Each logical device represents a COSEM application process, which is modelled by a particular set of COSEM objects.

In each physical device, there shall be at least one Management Logical Device implemented that may be the only logical device. It has a pre-defined Service Access Point, see Table 2. See also IEC 62056-5-3:2017, 4.1.9.

C.2.4 Application associations

C.2.4.1 Overview

As specified in IEC 62056-5-3:2017, 4.1.5, information exchange between clients and servers takes place within Application Associations (AA).

A client may be able to establish AAs with one or more servers. Likewise, a server may be able to support AA with one or more clients. Such AA may be simultaneously used or not, depending on the capabilities of the client and the server.

As specified in IEC 62056-5-3:2017, 4.1.5, AAs may be:

- explicitly established or pre-established;
- confirmed or unconfirmed;
- use short name (SN) referencing or logical name (LN) referencing to COSEM object attributes and methods with or without ciphering;
- use various authentication mechanisms;
- use various different xDLMS contexts;

The AAs are modelled with the Association objects, see IEC 62056-6-2:2017, 5.3.3. and 5.3.4. Each AA is identified by a pair of client and server SAPs.

C.2.4.2 Explicitly established AAs

Explicitly established AAs are established using the COSEM-OPEN service specified in IEC 62056-5-3:2017, 6.2. This involves an exchange of the ACSE AARQ and AARE APDUs to negotiate the application context, the authentication context and the xDLMS context. In the case of authentication using High Level Security an additional exchange is required to exchange the result of the processing of the challenge by the peer.

C.2.4.3 Pre-established AAs

As specified in IEC 62056-5-3:2017, 4.1.4, AAs may also be pre-established. This means that the contexts of the AA are pre-defined and known by the client and the server that wish to use them.

When using pre-established AAs, the exchanges to establish the AAs can be avoided. However, several features are not available:

- authentication of the peers using passwords or challenges;
- client user identification;
- exchanging public key certificates with the AARQ/ARRE exchange.

See also IEC 62056-5-3:2017, 6.2.

Table C.1 specifies a pre-established AA – that may be used, but not mandatory – between Client with SAP = 0x7D and the Management Logical Device with SAP = 0x01.

Table C.1 – Pre-established association parameters

Parameter	Value
Client SAP	0x7D
Server SAP	0x01
Application context name	Short_Name_Referencing_With_Ciphering joint-iso-ccitt(2) country(16) country-name(756) identified-organisation(5) DLMS-UA(8) application-context(1) context_id(4)}
xDLMS version	0x06
Client Maximum PDU Size	58
Server Maximum PDU Size	82
Conformance block	• general-protection (1) • general-block-transfer (2) • read (3) • write (4) • unconfirmed-write (5) • parameterized-access (18)

The objects visible in this association:

- shall contain the mandatory objects as specified in C.3.4;
- the utility table objects specified in Table C.3;
- the security related objects as specified in C.3.5, with their class_id, version, base_name and access rights;
- any other object required for the application with their class_id, version, base_name and access rights.

C.2.4.4 The mandatory public AA

One AA is mandatory: this is the public AA between the Public client and the Management Logical Device. It shall support an AA to a public client with an application context without ciphering and with lowest level security (no security) authentication. Its role is to support revealing the internal structure of the physical device and to support notification of events in the server. See also IEC 62056-6-2:2017, 4.8.5.

NOTE Notifications by the server are not supported in the communication profile specified in this document.

Since the public AA does not provide security it may be used for accessing non-sensitive information only.

C.2.4.5 Other AAs

In addition to the mandatory public AA, any other explicitly established or pre-established AAs may be specified using the application context, authentication context, xDLMS context and object_list as required.

C.3 The COSEM objects

C.3.1 COSEM object model

The COSEM object model is specified in IEC 62056-6-2:2017.

As specified in IEC 62056-6-2:2017, 4.1, a COSEM object is a collection of attributes and methods. Attributes represent the characteristics of an object. The value of an attribute can affect the behaviour of an object. The first attribute of any object is the logical_name. It is one part of the identification of the object. An object may offer a number of methods to either examine or modify the values of the attributes.

Objects that share common characteristics are generalized as an interface class, identified with a class_id. An interface class may have one or more versions.

To exchange data, attributes and methods of the COSEM objects have to be accessed.

C.3.2 Referencing methods

As specified in IEC 62056-5-3:2017, 4.2.4.3, there are two referencing methods (styles) available to reference COSEM object attributes and methods:

- short name (SN) referencing; and
- logical name (LN) referencing.

The client always uses LN referencing. The server may use LN or SN referencing (or both).

When LN referencing is used, COSEM object attributes and methods are accessed with the xDLMS service requests using LN referencing: the GET, SET, ACTION and ACCESS services. They are specified IEC 62056-5-3:2017, 6.6 to 6.9.

The LN service requests contain an attribute or method descriptor that is composed of the class_id, the logical_name (i.e. the OBIS code) and the attribute or method ID. An attribute or method descriptor is expressed on 9 bytes.

When SN referencing is used, each COSEM object attribute and method is mapped to a COSEM named variable, expressed on two bytes.

As specified in IEC 62056-5-3:2017 Clause 8, the short names are of type Integer16 with last three bits set to 0. With this, the range of short names is from 0x0008 to 0x FFF8 and there are 8 191 variable names available.

The logical_name attribute of each COSEM object instance is mapped to an implementation specific base_name that can be retrieved by reading the object_list attribute of the Association SN object(s). For some objects, the base_name is standardised. Base_names may also be specified in companion specifications.

The offsets for the other attributes and methods are specified in the COSEM interface class specifications.

COSEM object attributes and methods are accessed then with the xDLMS service requests using SN referencing: the Read or Write xDLMS services. They are specified in IEC 62056-5-3:2017, 6.12 to 6.14. The SN service requests include the base name.

As already mentioned above, the client always uses LN referencing. Therefore, if the server uses SN referencing the Client SN_Mapper Application Service Element (ASE) in the client maps the LN service primitives to SN service primitive and vice versa. The mappings are specified in IEC 62056-5-3:2017, 7.3.9 to 7.3.11.

Service requests using short name referencing are shorter than the ones using logical name referencing. However, this difference is there only in the service requests, as the attribute and method descriptors are present only in the request.

On the other hand, on the client side, a translation between APDUs using LN referencing and SN referencing has to be performed for each request and response.

Example: A GET-request APDU containing the COSEM-Attribute-Descriptor has to be translated to a Read.request APDU containing the variable name to which the COSEM-Attribute-Descriptor has been mapped.

C.3.3 The object_list

In order to communicate with the server, the client needs to know the method to reference the COSEM object attributes and methods in the service requests.

It also has to know:

- the list of COSEM objects accessible in each AA;
- in the case of SN referencing, the base_name of each object; and
- the access rights to each attribute and method.

This information can be read from the Association LN / Association SN objects; see IEC 62056-6-2:2017, 5.3.3 and 5.3.4.

The object_list may also be specified in project specific companion specifications.

C.3.4 Mandatory COSEM objects

As specified in IEC 62056-6-2:2017, 4.8.4, the following objects shall be present in each COSEM logical device. They shall be accessible for GET/Read in all AAs with this logical device:

- COSEM logical device name object;
- current Association object, "Association SN" or "Association LN" in line with the referencing method used.

If there are several logical devices, a SAP assignment object has to be present in the management Logical Device. In this case, the COSEM logical device name object does not have to be present.

The reserved base_names of the mandatory objects are specified in IEC 62056-6-2:2017, 4.3.

C.3.5 Application specific objects

The other COSEM objects may be implemented according to functionality required.

To support cryptographic protection of xDLMS APDUs the following objects are necessary:

- Security setup objects, see IEC 62056-6-2:2017, 5.3.7;
- Invocation counter objects, see IEC 62056-6-2:2017, 6.2.33.

To support COSEM data protection, Data protection objects, see IEC 62056-6-2:2017, 5.3.9.

C.3.6 Utility table objects

IEC 62056-6-2:2017, 5.2.7 specifies Utility table objects that allow encapsulating ANSI C12.19 Utility Industry End Device Data Tables AKA "Utility tables". Each Utility table is represented by an instance of this IC, identified by its logical name. The OBIS codes of Utility tables are specified in IEC 62056-6-2:2017, 6.2.36.

Using these Utility table objects provides a migration path from applications using the ANSI C12.19 tables to DLMS/COSEM: the Utility table objects may co-exist with other COSEM objects in the same or in different AAs and the data held in the tables can be accessed by DLMS/COSEM services.

ANSI C12.19 specifies:

- 2 048 Standard tables (table identifier range 0 to 2 047);
- 2 048 Manufacturer tables (table identifier range 2 048 to 4 095);
- 2 048 Standard pending tables (table identifier range 4 096 to 6 143);
- 2 048 Manufacturer pending tables (table identifier range 6 144 to 8 191).

There are also user-defined tables.

To use the tables with SN referencing, the attributes of the Utility table objects have to be mapped to short names. As each Utility table object has 4 attributes, mapping all attributes of Standard and Manufacturer tables to short names would require 32 640 named variables, but only 8 191 are available. This limits the number of tables/Utility table objects that can be accessed using SN referencing.

The base_names of the Utility tables implemented can be retrieved by reading the object_list attribute of the Association SN object that models the AA with the logical device implementing the tables.

To avoid reading the object_list, this document specifies base_names for selected Utility tables.

Table C.2 specifies the base_name ranges available for the four utility table categories.

Table C.2 – base_name range for tables categories

Table category	Start base name	End base name	Table Id	COSEM Objects	
				From	To
Standard Tables	0x9000	0x9760	0000 – 79	0-0:65.0.0.255	0-0:65.0.79.255
Manufacturer Tables	0x9800	0xAFF8	0000 – 2047	0-0:65.16.0.255	0-0:65.31.127.255
Standard pending Tables	0xB000	0xB7E0	0000 – 2047	0-0:65.32.0.255	0-0:65.47.127.255
Manufacturer Pending tables	0xB800	0xCFF8	0000 – 2047	0-0:65.48.0.255	0-0:65.63.127.255

Table C.3 defines the base_name for the first Standard tables, named BT00 up to BT79.

In the scope of this document, reading the object_list from the server can be avoided if it is known in advance, meaning:

- all these standard tables and only these tables are used;
- the base_name of all other tables is specified in the companion specification;

  NOTE It is assumed that each attribute of the Utility table objects is implemented and the access rights are specified in the companion specification.

- the base_name, interface_class version and access rights to each attribute and method of all other COSEM objects are specified in the companion specification.

Table C.3 – base_names for standard tables BT00 to BT79

Table ID	OBIS code	base_name	Name
BT00	0-0:65.0.0.255	0x9000	General Configuration
BT01	0-0:65.0.1.255	0x9020	General Manufacturer Identification
BT02	0-0:65.0.2.255	0x9040	Device Nameplate
BT03	0-0:65.0.3.255	0x9060	End Device Mode Status
BT04	0-0:65.0.4.255	0x9080	Pending Status
BT05	0-0:65.0.5.255	0x90A0	Device Identification
BT06	0-0:65.0.6.255	0x90C0	Utility Information
BT07	0-0:65.0.7.255	0x90E0	Procedure Initiate
BT08	0-0:65.0.8.255	0x9100	Procedure response
BT10	0-0:65.0.10.255	0x9120	Dimension Source Limiting
BT11	0-0:65.0.11.255	0x9140	Actual sources limiting
BT12	0-0:65.0.12.255	0x9160	Unit of Measure Entry
BT13	0-0:65.0.13.255	0x9180	Demand Control
BT15	0-0:65.0.15.255	0x91A0	Constants
BT16	0-0:65.0.16.255	0x91E0	Source definition
BT17	0-0:65.0.17.255	0x9200	Alternate source definition
BT20	0-0:65.0.20.255	0x9220	Dimension Register
BT21	0-0:65.0.21.255	0x9240	Actual Register
BT22	0-0:65.0.22.255	0x9260	Data Selection
BT23	0-0:65.0.23.255	0x9280	Current Register Data
BT24	0-0:65.0.24.255	0x92A0	Previous Season Data
BT25	0-0:65.0.25.255	0x92C0	Previous Demand Reset
BT26	0-0:65.0.26.255	0x92E0	Self-Read Data
BT27	0-0:65.0.27.255	0x9300	Present Register Selection
BT28	0-0:65.0.28.255	0x9320	Present Register Data
BT30	0-0:65.0.30.255	0x9340	Dimension Display
BT31	0-0:65.0.31.255	0x9360	Actual Display
BT32	0-0:65.0.32.255	0X9380	Display Source
BT33	0-0:65.0.33.255	0x93A0	Primary Display List
BT40	0-0:65.0.40.255	0x93C0	Dimension Security Limiting
BT41	0-0:65.0.41.255	0x93E0	Actual Security Limiting
BT42	0-0:65.0.42.255	0x9400	Security
BT43	0-0:65.0.43.255	0x9420	Default Access Control
BT44	0-0:65.0.44.255	0x9440	Access Control
BT45	0-0:65.0.45.255	0x9460	Key
BT50	0-0:65.0.50.255	0x9480	Dimension Time and TOU
BT52	0-0:65.0.52.255	0x94A0	Clock
BT53	0-0:65.0.53.255	0x94C0	Time Offset
BT54	0-0:65.0.54.255	0x94E0	Calendar
BT55	0-0:65.0.55.255	0x9500	Clock State
BT56	0-0:65.0.56.255	0x9520	Time Remaining Table
BT60	0-0:65.0.60.255	0x9540	Dimension Load Profile
BT61	0-0:65.0.61.255	0x9560	Actual Load Profile

Table ID	OBIS code	base_name	Name
BT62	0-0:65.0.62.255	0x9580	Load Profile Control
BT63	0-0:65.0.63.255	0x95A0	Load Profile Status
BT64	0-0:65.0.64.255	0x95C0	Load Profile Data 1
BT65	0-0:65.0.65.255	0x95E0	Load Profile Data 2
BT66	0-0:65.0.66.255	0x9600	Load Profile Data 3
BT67	0-0:65.0.67.255	0x9620	Load Profile Data 4
BT70	0-0:65.0.70.255	0x9640	Dimension Log
BT71	0-0:65.0.71.255	0x9660	Actual Log
BT72	0-0:65.0.72.255	0x9680	Event Identification
BT73	0-0:65.0.73.255	0x96A0	History Log Control
BT74	0-0:65.0.74.255	0x96C0	History Log Data
BT75	0-0:65.0.75.255	0x96E0	Event Log Control
BT76	0-0:65.0.76.255	0x9700	Event Log Data
BT77	0-0:65.0.77.255	0x9720	Event Log and Signatures Enable
BT78	0-0:65.0.78.255	0x9740	End Device Program State Table
BT79	0-0:65.0.79.255	0x9760	Event Counters Table

Annex D
(informative)

OSGP-AES-128-PSK security suite

D.1 General

The OSGP-AES-128-PSK security suite provides a protected two-way communication channel between a NNAP (DC) and an LNAP (meter). The secure channel provides data confidentiality, data and origin integrity, and replay protection. It can be used for both broadcast communication and unicast communication.

OSGP-AES-128-PSK is specifically designed for constrained-networks of legacy low-end embedded devices.

The security suite consists of three logical layers (1 to 3) as shown in Table D.1.

Table D.1 – Security logical layers

Logical layer	Name	Contents
4	ADP	Upper layer protocol
3	Secure Channel Protocol	Secure channel initialization Secure channel unicast communication Secure channel broadcast communication
2	Cryptographic Functions	OSGP_KDF OSGP_MAC OSGP_MAC_VERIFY OSGP_AE OSGP_AD
1	Cryptographic Primitives	AES-128 CCM CMAC

Each logical layer uses the service of the logical layer beneath it (if any) and provides a service for the logical layer above it (if any). For example, the Secure Channel Protocol logical layer uses the OSGP_AE function provided by the Cryptographic Functions logical layer for protecting messages, which in turn is used to provide secure channel communication to the Adaptation logical layer above.

OSGP-AES-128-PSK requires that an initial pre-shared key has been securely established. How this is done, is outside the scope of this specification.

The specification of OSGP-AES-128-PSK is structured, based on the reverse order of the logical layers. Specifically, the remainder of this Annex is structured into the following top-level clauses:

- Background: sets the stage for the security suite by describing the target system, the threat model, and the design goals for the security suite;
- Terms and notation: describes the terms and notation used throughout this specification;
- Cryptographic primitives: specifies the low-level cryptographic primitives that act as building blocks for the security suite;

- Cryptographic functions: specifies the high-level cryptographic functions referenced in the protocol specification;
- Keys: specifies the keys and the key hierarchy in the cryptographic system;
- Secure channel initialization: specifies how a secure channel is established;
- Secure channel communication: specifies how the secure channel is used to protect network communication;
- Key management: specifies how keys are generated, distributed and renewed, and the recommended key validity periods;
- Error messages: specifies the common security-related error messages used to indicate failures in the protocol;
- Security considerations: provides additional recommended practices with respect to this security suite as well as reasoning behind the security suite's design;
- References: lists the literature cited throughout this document.

D.2 Background

D.2.1 Overview

This clause provides background information about the target system, the threat model considered, the design goals, and the inspiration for this security suite.

D.2.2 System Assumptions

The system assumption is a list of important assumptions about the target system.

- The target system is a network of legacy embedded devices that are 10 to 15 years old. Thus, LNAPs and NNAPs are expected to be significantly resource-constrained, both in terms of memory and computing power.
- The communication network is bandwidth-constrained, throughput-constrained, noisy, and substantially prone to data corruption during transfers, thus often requiring retries to complete a transaction.
- The communication medium is shared between a subnet of nodes (up to 1 024) and can thus only be used by one node at a time. Attempts at concurrent communication results in packet corruption.
- The underlying transport protocol cannot be modified.
- To address the problem of the shared medium, the NNAP is the only initiator of communication. That way, under normal network activity, a NNAP is able to control the use of the medium.

D.2.3 Threat Model

This security suite is concerned with a network adversary that has complete control over the communication network and knows everything about the cryptographic system except for the secret keys.

The adversary is assumed to have access to high-performance computing resources.

Although mounting man-in-the-middle attacks on a shared medium network may be difficult in practice, the adversary is assumed to be able to record, tamper, drop, and/or inject messages arbitrarily at any point in time in the protocol.

It is important to pay attention to the fact that the adversary is under the same constraints imposed by the low-performing network, just like all the other nodes on the network.

D.2.4 Design goals

The goals of this security suite are as follows:

- **Security**: the security suite should be conservative in its cryptographic design and follows modern best practices and recommendations from cryptographers and security experts. The security should be defined, as a minimum, by the secrecy of the cryptographic keys used.
- **Secure communication**: the security suite should provide:
  a) **Message confidentiality**: it should not be feasible to read any secure communication unless you have access to the (encryption) key being used.
  b) **Message origin integrity (authenticity):** it should be possible to verify the origin of a given message based on a shared secret.
  c) **Message data integrity**: it should be possible to verify that a given message has not changed since it was generated based on a shared secret.
  d) **Message replay prevention**: it should be possible to detect and reject replayed (old) messages.
  e) **Message reflection prevention**: it should be possible to detect and reject messages that are being redirected from its original destination to a new destination.
- **LNAP compartmentalization**: the security suite should provide means to limit the impact of LNAP (key) compromises as much as possible.
- **Developer friendly**: it should be straightforward for developers to program a secure, efficient, and relatively small implementation of the security suite.
- **Suitable for legacy, resource-constrained devices on slow communication networks**:
  f) **Minimal implementation footprint**: the size of the security suite implementation should be small compared to other security solutions with similar design goals while still achieving acceptable speeds. This is to cater for the memory-constrained devices in the target system.
  g) **Minimal computational overhead**: the security suite should run at acceptable speeds on 10 to 15-year-old embedded devices. That is, the computational overhead of using this security suite should not degrade the performance to unacceptable levels.
  h) **Minimal network overhead**: the security suite should reduce the cryptographic overhead per network packet and the number of packets needed as much as possible to cater for the possibility of a slow and unreliable target system network.

D.2.5 Inspiration

This security suite is to a great extent based on the Pre-Shared Key Extensible Authentication Protocol (EAP) IETF RFC 5433 as specified in IEEE Std 802.11ac-2013. Both of these established protocols share similar design goals with OSGP-AES-128-PSK. Note that CCMP-128 is largely considered secure in the industry (e.g. CCMP-128's use is included in NIST's Suite B Cryptography as of 2015).

D.3 Terms, definitions, abbreviated terms and notation

D.3.1 Terms and definitions

D.3.1.1
associated data
data that is authenticated but not encrypted

D.3.1.2
byte
string of 8 bits

D.3.1.3
CBC-MAC
Cipher Block Chaining-Message Authentication Code

Note 1 to entry: This note applies to the French language only.

D.3.1.4
CCM
Counter with Cipher Block Chaining-Message Authentication Code

Note 1 to entry: This note applies to the French language only.

D.3.1.5
ciphertext
encrypted payload followed by a message authentication code (MAC) protecting both the header and payload. The MAC is also used to verify contextual inputs which depend on the cryptoContext (see D.7.2)

D.3.1.6
DC
data concentrator

Note 1 to entry: In the client-server network model, a DC is a NNAP acting as the client. In the scope of this communication profile, a DC will never process messages that are not responses to its request.

Note 2 to entry: This note applies to the French language only.

D.3.1.7
GCM
Galois/Counter Mode

Note 1 to entry: This note applies to the French language only.

D.3.1.8
LNAP
smart meter

Note 1 to entry: In the client-server network model, this is the server. By design, in the scope of this communication profile, a LNAP will never send a request to the NNAP.

Note 2 to entry: This note applies to the French language only.

D.3.1.9
message
adaptation-layer protocol data unit (AdpPDU)

Note 1 to entry: It consists of a header followed by a payload. During transmission, when the protection is required, the payload is substituted by message-specific and context-specific ciphertext.

D.3.1.10
header
supplemental data placed at the beginning of a message which is authenticated but not encrypted (part of the associated data)

D.3.1.11
payload
actual intended information which is authenticated and encrypted when the protection is required

D.3.1.12
nonce
value that is used only once within a specified context

D.3.1.13
unicast
communication between one NNAP and one LNAP

D.3.1.14
transaction
(unicast) transaction consists of a single NNAP request message and a single LNAP response message

D.3.1.15
broadcast
one-way communication from one NNAP to a set of LNAPs

D.3.1.16
domain
network of LNAPs managed by a NNAP

Note 1 to entry: A domain can only be managed by a single NNAP at a time.

D.3.2 Abbreviated terms

Abbreviated term	Meaning
BCK	Broadcast Communication Key
DMK	Domain Master Key
HSK	Handshake Key
MCK	Meter Commissioning Key
MFWK	Meter Firmware Key
MMK	Meter Master Key
MUK	Meter Unique Key
MUKHSK	MUK-derived Handshake Key
MUKMCK	MUK-derived Meter Commissioning Key
UCK	Unicast Communication Key

D.3.3 Notation

A||B	The concatenation of two bit strings A and B
0x	The prefix to a bit string that is represented as hexadecimal characters
{...}	... is encrypted and authenticated.
"..."	... is an ASCII encoded string of bytes, e.g., "ABC" is the same as 0x414243
m[:n]	The n leftmost bytes of the byte string m, e.g., "ABC"[:2] is the same as "AB"
a.x	The x field in a
UINT*n*	Unsigned integer consisting of *n* bits
UINT*n*(*x*..*y*)	The bit range starting from bit index *x* to and including bit index *y* in UINT*n*. Bit index 0 is always the most significant bit (MSB).
FILL*n*(*x*..*y*)	Fill the bit range starting from bit index *x* to and including bit index y in a string of *n* bits with zeros. Bit index 0 is always the most significant bit (MSB).
ARRAY[*n*] OF *x*	An array consisting of *n* elements of *x*.

D.3.4 Other conventions

- The words "shall", "shall not", "required", "shall", "shall not", "should", "should not", "recommended", "may", and "optional" in this document are to be interpreted as described in IETF RFC 2119.

- Integer (types) shall be sent in network order (i.e. big-endian).
- Zero-numbering is used for bit and byte indices and offsets.

D.4 Cryptographic Primitives

D.4.1 General

At its lowest level (see Table D.1), the security suite is built upon three cryptographic primitives:

a) AES-128: the 128-bit key variant of the FIPS-approved symmetric block cipher algorithm, as specified in FIPS PUB 197.

b) CMAC: Cipher-based Message Authentication Code (CMAC), as specified in SP800-38B.

c) CCM: Counter with CBC-MAC (CCM), as specified in SP800-38C.

CMAC and CCM are generic algorithms that take a set of parameters. The remainder of Clause D.4 specifies the parameters used in this security suite.

Refer to Clause D.13 for the reasoning behind the choice of the three cryptographic primitives.

NOTE Both applications of AES-128 (CCM and CMAC) only use the encryption algorithm of AES-128, not the decryption algorithm. As a result, no code for anything related to AES-128 decryption is used, which significantly reduces the code space requirement.

D.4.2 CMAC

CMAC is a generic message authentication code algorithm specified in SP800-38B. CMAC has three main parameters: an underlying block cipher (CIPH) with a certain block size (b) and a MAC length (T_{len}). In this security suite, the CMAC parameters are set as follows:

- CIPH = AES-128; the underlying block cipher algorithm is AES-128 as specified in FIPS PUB 197.
- b = 128 (bits); the block size of CIPH is 128 bits (this is the standard block size of AES-128).
- T_{len} = 128 (bits); the length of the output MAC is 128 bits.

From this point on, this configuration of CMAC is referred to as AES-128-CMAC.

D.4.3 CCM

Counter with CBC-MAC (CCM) is a generic authenticated block cipher mode of operation first specified in IETF RFC 3610 and later in SP800-38C CCM has three primary parameters: the length of the MAC (t), the byte length of the binary representation of the byte length of the payload (q), and the underlying block cipher. In this security suite, the CCM parameters are set as follows:

- t = 8; the MAC is 8 bytes (64 bits).
- q = 2; the length field is 2 bytes (payload length is thus limited to 65 535 bytes which is more than enough to hold the largest possible payload).
- Block cipher = AES-128; the underlying block cipher used is AES-128 as specified in FIPS PUB 197.

As a result of these values, the input nonce length (n) is 13 bytes (as per the n+q = 15 constraint). In the remainder of this Annex, AES-128-CCM refers to CCM with the above configuration as specified in SP800-38C.

Refer to clause D.13.1.2 for the reasoning behind choosing an 8-byte MAC (t).

NOTE This is the same configuration of CCM that is used in the CCMP-128 as specified in IEEE Std 802.11ac-2013

D.5 Cryptographic Functions

D.5.1 Overview

This clause specifies the cryptographic functions used in the remainder of the specification. They use the cryptographic primitives specified in Clause D.4 as building blocks. The cryptographic functions in this security suite are as follows:

- OSGP_KDF: a key derivation function.
- OSGP_MAC: a message authentication code function.
- OSGP_MAC_VERIFY: a message authentication code verification function.
- OSGP_AE: an authenticated encryption function.
- OSGP_AD: an authenticated decryption function.
- OSGP_CSPRNG: a cryptographically strong pseudo random number generator.

D.5.2 OSGP_KDF: Key Derivation Function

The OSGP_KDF is an instance of the generic key derivation function as specified by SP 800-108. It uses the AES-128-CMAC construction as the underlying pseudo random function (PRF) as specified in SP 800-108. It is used to derive keys from a key derivation key in the cryptographic system. It aims to make it infeasible to go from a derived key back to the root key. The OSGP_KDF is specified in pseudo-code below (refer to SP 800-108 for more information).

OSGP_KDF(input_key, context, label)

Inputs:

- input_key: 16 bytes of input key material.
- context: 6 bytes of contextual information. This is a meter-unique identifier when deriving meter-unique keys and a domain-unique identifier when deriving domain-unique keys.
- label: a 3-byte key-type-unique string.

Output:

- 16 bytes of output key material.

Steps:

1. data = 0x01 || label || 0x00 || context || 0x80
2. output_key = AES_128_CMAC(message=data, key=input_key)
3. return output_key

As specified above, the OSGP_KDF function takes three arguments: input key material (input_key), contextual information (context) and a key-type-unique label (label). Internally, this function combines the label and the context as specified in SP 800-108 and uses the result as the message input to the AES-128-CMAC construction (specified in SP800-38B) using the input_key as the 16-byte input key material. OSGP_KDF returns 16 bytes of derived keying material which can be used for a new key.

D.5.3 OSGP_MAC: Message Authentication Code Function

The OSGP_MAC function is used to generate message authentication codes that can be used to verify the authenticity of the given message using a pre-shared key. It is based on the AES-128-CMAC construction as specified in SP800-38B.

The OSGP_MAC function is specified below.

OSGP_MAC(data, key)

Inputs:

- data: an n-byte message where $0 \leq n < 2^{32}$
- key: a 16-byte key

Output:

- 16-byte message authentication code (MAC)

Steps:

1. mac = AES_128_CMAC(message=data, key=key)
2. return mac

AES_128_CMAC refers to the "MAC Generation" process as specified in SP800-38B.

D.5.4 OSGP_MAC_VERIFY: Message Authentication Code Verification Function

The OSGP_MAC_VERIFY is used to verify a message authentication code generated by OSGP_MAC. It is specified below.

OSGP_MAC_VERIFY(data, key, external_mac)

Inputs:

- data: an n-byte message where $0 \leq n < 2^{32}$
- key: a 16-byte key
- external_mac: a 16-byte message authentication code to verify

Output:

- Either VALID or INVALID.

Steps:

1. internal_mac = OSGP_MAC(data=data, key=key)
2. Perform a bit-wise, constant-time comparison of internal_mac and external_mac. If they contain the same exact string of bits, then return VALID; otherwise return INVALID

As specified above, the verification of the given MAC (external_mac) is simply done by first recomputing it (internal_mac) and then comparing the two MACs.

If OSGP_MAC_VERIFY returns INVALID, then the message is not authentic.

If OSGP_MAC_VERIFY returns VALID, then the message should be authentic as per the assurance provided by the underlying AES-128-CMAC construction.

It is critical that the bit-wise comparison of the given MAC (external_mac) and the recomputed mac (internal_mac) is done in constant time (step 2). If not, then it may be possible for the network adversary to determine the correct (internal_mac) MAC for a message by measuring the timing differences of different external_mac guesses. This is commonly referred to as a timing side-channel attack.

D.5.5 OSGP_AE/OSGP_AD: Authenticated Encryption/Decryption Function

D.5.5.1 Overview

The OSGP_AE and OSGP_AD functions are used for authenticated encryption and authenticated decryption, respectively.

D.5.5.2 OSGP_AE

The OSGP_AE function uses the "Generation-Encryption" process as specified in SP800-38C. This is referred to as the AES_128_CCM_generation_encryption function in the OSGP_AE specification below. Refer to SP800-38C for details.

OSGP_AE(payload, associated_data, nonce, key)

Inputs:

- payload: an n-byte payload where $0 \leq n < 2^{16}$. This byte string will be authenticated and encrypted.
- associated_data: a-byte associated data where $0 \leq a < 30$. This byte string will be authenticated but not encrypted.
- nonce: a 13-byte nonce.
- key: a 16-byte key.

Output:

- $n + t$ bytes of ciphertext where n is the number of bytes of the input payload and t is the length of the MAC as specified in D.4.3.

Steps:

1. ciphertext = AES_128_CCM_generation_encryption(
   P=payload,
   N=nonce,
   A=associated_data,
   K=key)
2. return ciphertext

As specified in SP800-38C, the ciphertext is the encrypted payload immediately followed by the MAC which is t bytes long (t is specified in D.4.3). The MAC is computed over the associated data and the payload.

As required by CCM, the nonce shall never repeat under the same key. This is ensured by the nonce creation schemes specified in D.8.3.3 and D.8.4.3.

D.5.5.3 OSGP_AD

The OSGP_AD uses the "Decryption-Verification" process as specified inSP800-38C. This is referred to as the AES_128_CCM_decryption_verification function in the OSGP_AD specification below. Refer to SP800-38C for details.

OSGP_AD(ciphertext, associated_data, nonce, key)

Inputs:

- ciphertext: c-byte purported ciphertext where $t \leq c < 2^{16} + t$ and t is specified in D.4.3.
- associated_data: associated data (see OSGP_AE specification for more information).
- nonce: nonce (see OSGP_AE specification for more information).
- key: a 16-byte key

Outputs:

- status: VALID or INVALID.
- payload: if status is VALID, then the n-byte payload output of OSGP_AE; otherwise (status is INVALID) a zero-byte output.

Steps:

1. payload_or_invalid = AES_128_CCM_decryption_verification(
   C=ciphertext,
   A=associated_data,
   N=nonce,
   K=key)
2. if payload_or_invalid == INVALID:
   status = INVALID
   payload = ""
   else:
   status = VALID
   payload = payload_or_invalid
3. return status, payload

When implementing the AES_128_CCM_decryption_verification function (i.e. the Decryption-Verification process in SP800-38C), attention needs to be paid to the comparison of the two MACs in constant time (item 10 of 6.2, subclause Steps in SP800-38C).

D.5.6 OSGP_CSPRNG: Cryptographically Secure Pseudo Random Number Generator

The OSGP_CSPRNG function is used to generate random numbers suitable for cryptographic purposes.

This document does not define a specific underlying cryptographically secure pseudo random number generator (CSPRNG). It is therefore up to the implementer to choose a suitable CSPRNG.

In the specification of OSGP_CSPRNG below, the underlying CSPRNG is referred to as the platform_csprng(n) function, which returns n random bytes suitable for cryptographic use.

OSGP_CSPRNG(num_of_bytes)

Input:

- num_of_bytes: the number of random bytes to generate.

Output:

- Random bytes (num_of_bytes) suitable for cryptographic purposes.

Steps:

1. If num_of_bytes <= 0:
   return ""
2. random_bytes = platform_csprng(num_of_bytes)
3. return random_bytes

D.6 Keys

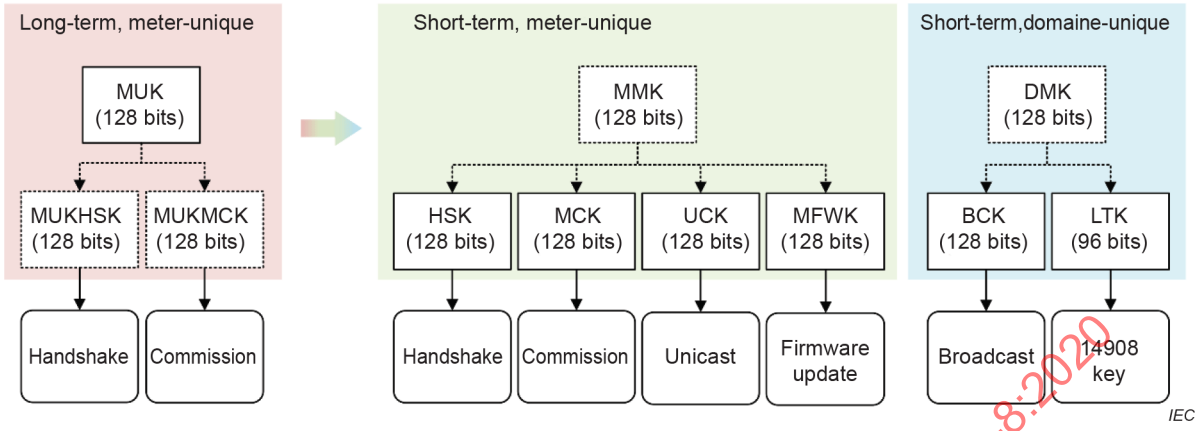

Figure D.1 – Key hierarchy diagram

Figure D.1 above presents an overview of the keys used in the cryptographic system. Each square box represents a key given its unique identifier and length in bits. A dotted arrow indicates a one-way key derivation. For example, the UCK is derived from the MMK in a way that makes it infeasible to go from the UCK back to the MMK. The dotted line around the MMK indicates that the MMK is a temporary key derivation key that should be securely erased by both the NNAP and LNAP after using it to derive its sub-keys. The same is true for the DMK except that only the LNAP should securely erase it after deriving its sub-keys, not the NNAP (it needs it for commissioning new LNAPs as specified in Clause D.7).

Similarly, the dotted lines around the MUK sub-keys indicate that they should be erased after the first use (as specified in Clause D.7.) A rounded square indicates a process. For example, "Commission" represents the commission process as specified in Clause D.7. A solid arrow connecting a key to a process means that the key is used to secure that specific process. For example, the HSK protects the handshake transaction during secure channel initialization as specified in Clause D.7.

In the diagram, each key is grouped based on its validity period and its scope. The red, green, and blue groupings consist of long-term meter-unique keys, short-term meter-unique keys, and short-term domain-unique keys, respectively.

There is one exception: the long-term keys that are derived from the MUK should be erased after use.

The long-term keys are used to (re-)establish the initial trust between a LNAP and NNAP. Once done, the initial trust is used to securely establish short-term keys, which are then used from that point on (indicated by the arrow going from the long-term keys to the short-term keys). If the short-term keys are suspected to be compromised or for some reason no longer accessible, then the long-term keys can be used instead.

See Clause D.10 for more information about key management.

See Table D.2 for a detailed definition of each key (size is in bits).

Table D.2 – Key type and their scope

	Size	Origin	Transferred	Purpose	Validity period	Scope	Access
MUK	128	CSPRNG	Outside the scope of this specification	Key derivation (Clause D.7.)	Long-term	Meter-unique	LNAP, NNAP
MUKHSK	128	Derived from MUK	No	Handshake (Clause D.7.)	Used once, then erased	Meter-unique	LNAP, NNAP
MUKMCK	128	Derived from MUK	No	Commission (Clause D.7.)	Used once, then erased	Meter-unique	LNAP, NNAP
MMK	128	CSPRNG	Commissioning request	Key derivation (Clause D.7.)	Erased after deriving its sub-keys	Meter-unique	LNAP, NNAP
HSK	128	Derived from MMK	No	Handshake (Clause D.7.)	Short-term	Meter-unique	LNAP, NNAP
MCK	128	Derived from MMK	No	Commission (Clause D.7.)	Short-term	Meter-unique	LNAP, NNAP
UCK	128	Derived from MMK	No	Unicast communication (Clause D.8).	Short-term	Meter-unique	LNAP, NNAP
MFWK	128	Derived from MMK	No	Firmware update (Clause D.9).	Short-term	Meter-unique	LNAP, NNAP
DMK	128	CSPRNG	Commissioning request	Key derivation (Clause D.7.)	Erased after deriving its sub-keys	Domain-unique	Domain, NNAP
LTK	96	Derived from DMK	No	ISO/IEC 14908-1 authentication	Short-term	Domain-unique	Meter, NNAP
BCK	128	Derived from DMK	No	Broadcast communication (Clause D.8).	Short-term	Domain-unique	Domain, NNAP

A "LNAP, NNAP" access means that the particular key is shared between one LNAP and one NNAP whereas a "Domain, NNAP" access means that the key is shared between one NNAP and all the LNAPs on its specific domain.

As specified above, the following keys shall be generated using a cryptographically strong pseudorandom number generator (CSPRNG) suitable for generating cryptographic keys:

- MUK
- MMK
- DMK

The rest of the keys in the cryptographic system are keys derived from either the MUK, MMK, or the DMK.

How the MUK is securely pre-shared between the LNAP and the NNAP is outside the scope of this specification.

D.7 Secure channel initialization

D.7.1 Overview

Before being able to securely communicate on the network, the NNAP needs to establish a cryptographic context (CryptoContext) with its set of managed LNAPs on its domain. A CryptoContext is established on a per-LNAP basis using pre-shared meter-unique keys for authentication and encryption.

This clause specifies what a CryptoContext is and how it is established between a NNAP and a LNAP.

D.7.2 Secure Channel State (CryptoContext)

A CryptoContext contains the following fields:

- HSK: handshake key, see Clause D.6;
- MCK: meter commissioning key, see Clause D.6;
- UCK: unicast communication key, see Clause D.6;
- MFWK: meter firmware key, see Clause D.6;
- BCK: broadcast communication key, see Clause D.6;
- LTK: ISO/IEC 14908-1 key, see Clause D.6;
- USeq: 4-byte request sequence number used for unicast communication;
- BSeq: 4-byte domain-unique broadcast sequence number used for broadcast communication;
- URand: 2 randomly generated bytes;
- BRand: 5 randomly generated bytes;
- DomainID: a 3-byte network domain_ID;
- SubnetID: a 1-byte network subnet_ID;
- NodeID: a 1-byte network node_ID;
- MID: a 6-byte meter-unique ID.

Everything except USeq, BSeq, URand, BRand, DomainID, SubnetID, NodeID, and MID is secret information that shall be protected from unauthorized access and securely erased when no longer needed. Although the URand and BRand are never transmitted in the clear, they do not have to be treated as secret information. See D.8.3 and D.8.4 for what the URand and BRand are used for, respectively.

The MID is assumed to be known to both the NNAP and the LNAP before the first initialization. How the MID is pre-shared prior to the initialization is outside the scope of this specification. Note that the MID is not a secret.

A CryptoContext is valid until the next initialization. When a new CryptoContext has been established, then all sensitive information from the old CryptoContext shall be securely erased and replaced with new values.

D.7.3 Flow

This subclause specifies how a CryptoContext used for the secure channel is (re-)initialized between a NNAP and a LNAP. An overview of the message flow is shown in Figure D.2 (see Clause D.2):

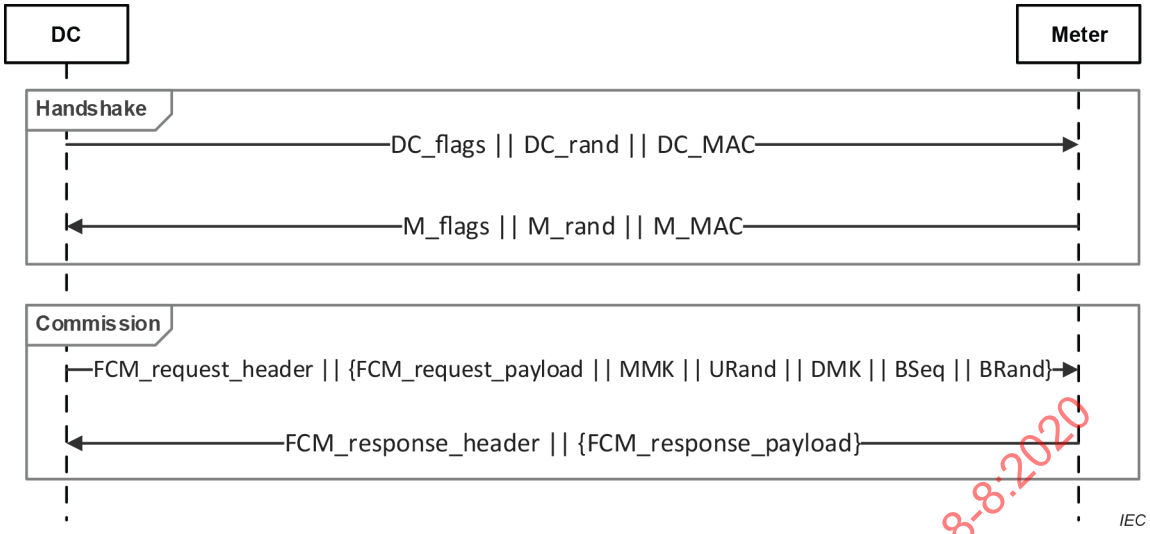

Figure D.2 – Secure channel establishment

Two transactions are needed to complete the process (one transaction consists of 1 request message and 1 response message, so 4 messages in total). The first transaction is referred to as the "handshake" transaction which consists of a challenge-response flow that ensures freshness and sets up the process for mutual authentication. The second transaction is referred to as the "commission" transaction which configures the LNAP and (re-)initializes the CryptoContext shared between the NNAP and the LNAP. The two transactions are cryptographically tied to each other making it infeasible to splice or otherwise combine or replay past flows without it being detected by the NNAP and/or LNAP.

The formats of the different message types are specified in D.7.7. Error messages are specified in Clause D.12.

The process specification uses the cryptographic functions specified in Clause D.5. They are prefixed with OSGP_.

The entire process is specified below.

Let:

- handshake_key: the handshake key used for authenticating the handshake messages. This is the HSK from the current CryptoContext or the MUKHSK. The MUKHSK is derived as follows:
  OSGP_KDF(key=MUK, context=MID, label="hsk");
- commission_request_key: the key used for protecting the commissioning request (CommissionRequest). This is the MCK from the current CryptoContext or the MUKMCK. The MUKMCK is derived as follows:
  OSGP_KDF(key=MUK, context=MID, label="mck");
- commission_response_key: the key used for protecting the commissioning response (CommissionResponse). This is the new MCK where *new* means that it is a key derived from the new MMK found in the CommissionRequest;
- MID: the meter-unique identifier for the LNAP in question;
- BSeq: the current value of the NNAP's broadcast sequence number;
- BRand: current random data for secure broadcast (NNAP only);
- DMK: the current DMK for the domain (NNAP only).

Result of a successful process:

- A meter-unique CryptoContext shared between the NNAP and the LNAP, which can be used for secure communication as specified in Clause D.8.

Steps:

Handshake transaction

1) NNAP constructs ChallengeRequest message:
2) DC_flags =0x061001
3) DC_rand =OSGP_CSPRNG(16)
4) DC_MAC =OSGP_MAC(data=MID || DC_flags || DC_rand, key=handshake_key)
5) NNAP sends the ChallengeRequest to LNAP as per Figure D.3:

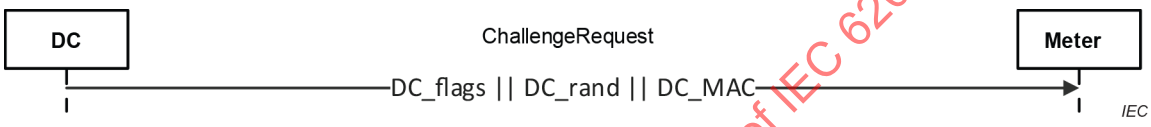

Figure D.3 – Challenge request

6) LNAP receives ChallengeRequest
7) LNAP verifies DC_MAC:
status = OSGP_MAC_VERIFY(data=MID || DC_flags || DC_rand, key=handshake_key, external_mac=DC_MAC)
8) If status is INVALID, then LNAP responds with an AuthenticationFailure and aborts the process; otherwise, LNAP proceeds.
9) LNAP constructs a ChallengeResponse message:
10) M_flags = 0x0600ssoo (ss is the supported security modes value and oo is the optional features bitmask, see D.7.7.3, for details).
11) M_rand = OSGP_CSPRNG(16)
12) M_MAC = OSGP_MAC(data=MID || M_flags || M_rand || DC_rand, key=handshake_key)
13) LNAP sends ChallengeResponse to NNAP as per Figure D.4

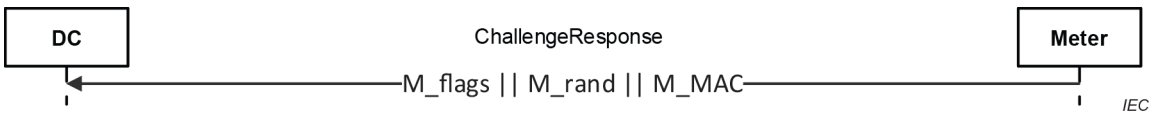

Figure D.4 – Challenge response

14) NNAP receives ChallengeResponse
15) NNAP verifies M_MAC:
status = OSGP_MAC_VERIFY(data=MID || M_flags || M_rand || DC_rand, key=handshake_key, external_mac=M_MAC)
16) If status is INVALID, then the NNAP should log the event and shall abort the process; otherwise, NNAP proceeds with the process.

Commission transaction

17) NNAP constructs a CommissionRequest:

18) FCM_request_header = see D.7.7.4

19) FCM_request_payload = see D.7.7.4

20) MMK = OSGP_CSPRNG(16)

21) URand = OSGP_CSPRNG(2)

22) NNAP constructs a 13-byte request nonce:
nonce = 0x02 || DC_rand[:6] || M_rand[:6]

23) NNAP constructs associated data for the CommissionRequest message:
associated_data = M_mac || FCM_request_header

24) NNAP performs an authenticated encryption of the CommissionRequest:
ciphertext =OSGP_AE(
payload=FCM_request_payload || MMK || URand || DMK || BSeq || BRand,
associated_data=associated_data,
nonce=nonce,
key=commission_request_key)

25) NNAP sends FCM_request_header || ciphertext (a protected CommissionRequest) to LNAP as per Figure D.5:

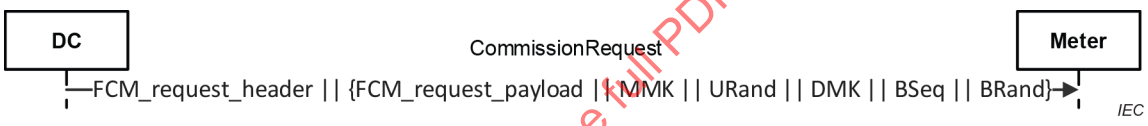

Figure D.5 – Commission Request

26) LNAP receives the protected CommissionRequest (header || ciphertext)

27) LNAP constructs the request nonce as done by the NNAP during step 10:
nonce = 0x02 || DC_rand[:6] || M_rand[:6]

28) LNAP constructs associated data as done by the NNAP during step 11:
associated_data = M_mac || FCM_request_header

29) LNAP performs an authenticated decryption of the protected CommissionRequest:
status, payload = OSGP_AD(ciphertext=ciphertext,
associated_data=associated_data,
nonce=nonce,
key=commission_request_key)

30) If status is INVALID, then LNAP responds with an AuthenticationFailure message and aborts the process; otherwise LNAP proceeds (payload is the decrypted ciphertext).

31) LNAP updates its internal CryptoContext based on the values in the authenticated and decrypted CommissionRequest:

a. CryptoContext.HSK = OSGP_KDF(
input_key=CommissionRequest.MMK,
context=MID,
label="hsk")

b. CryptoContext.MCK = OSGP_KDF(
input_key=CommissionRequest.MMK,
context=MID,
label="mck")

c. CryptoContext.UCK = OSGP_KDF(
input_key=CommissionRequest.MMK,
context=MID,
label="uck")

d. CryptoContext.MFWK = OSGP_KDF(
input_key=CommissionRequest.MMK,
context=MID,
label="fwk")
e. CryptoContext.BCK = OSGP_KDF(
input_key=CommissionRequest.DMK,
context=CommissionRequest.DomainID || 0x000000,
label="bck")
f. CryptoContext.LTK = OSGP_KDF(
input_key=CommissionRequest.DMK,
context=CommissionRequest.DomainID || 0x000000,
label="ltk")[:12]
g. CryptoContext.USeq = 1
h. CryptoContext.URand = CommissionRequest.URand
i. CryptoContext.BRand = CommissionRequest.BRand
j. CryptoContext.BSeq = CommissionRequest.BSeq
k. CryptoContext.DomainID = CommissionRequest.DomainID
l. CryptoContext.SubnetID = CommissionRequest.SubnetID
m. CryptoContext.NodeID = CommissionRequest.NodeID

32) LNAP makes sure that the overwritten values of its CryptoContext are securely erased.

33) LNAP securely erases MMK and DMK.

34) LNAP updates its configuration as specified in D.7.5.

35) LNAP constructs a 13-byte response nonce:
nonce = 0x03 || DC_rand[:6] || M_rand[:6]

36) LNAP constructs associated data for the response:
associated_data =
nonce ||
CryptoContext.DomainID || CryptoContext.SubnetID || CryptoContext.NodeID ||
fcm_response_header

37) LNAP constructs and performs an authenticated encryption of the CommissionResponse:
ciphertext =OSGP_AE(payload=fcm_response_payload,
associated_data=associated_data,
nonce=nonce,
key=commission_response_key)

38) LNAP sends FCM_response_header || ciphertext (a protected CommissionResponse) to NNAP as per Figure D.6.

Figure D.6 – Commission response

39) NNAP receives a protected CommissionResponse

40) NNAP constructs the 13-byte response nonce:
nonce = 0x03 || DC_rand[:6] || M_rand[:6]

41) NNAP constructs the associated data as done by the LNAP in step 37:
associated_data =
nonce ||
CryptoContext.DomainID || CryptoContext.SubnetID || CryptoContext.NodeID ||
FCM_response_header

42) NNAP performs an authenticated decryption of the CommissionResponse:
payload, status = OSGP_AD(ciphertext=ciphertext,
associated_data=associated_data,
nonce=nonce,
key=commission_response_key)

43) If status is INVALID, then NNAP should log the event and shall abort the process; otherwise, the NNAP proceeds.

44) NNAP updates the per-LNAP CryptoContext exactly like the LNAP (step 24).

45) NNAP securely erases the MMK.

46) NNAP should log the event of a successful process.

Note that in some deployments it may be preferred to use the MUK for all commissioning requests. However, for systems where the MUK has a very long lifetime or where the MUK cannot be updated, using this key for frequent key updates may be undesirable. Ultimately, this is a decision that should be made by the user. This protocol simply allows for different approaches to cater to different needs.

As specified above, the MMK shall be freshly generated on each initialization. This ensures key independence and practical uniqueness and is an important part of the sequence number schemes. Unicast sequence numbers (USeq) shall reset to 1 during commissioning. Otherwise, re-using an MMK would result in re-using one or more sequence numbers which in turn will break the assumptions of sequence numbers being increasing, a property that both the nonce and replay prevention schemes critically rely on.

If it is undesirable to use the long term MUK, then the NNAP shall keep track of both the current and the new CryptoContext until it receives a valid response to its CommissionRequest. A valid response is a CommissionResponse. Not receiving a CommissionResponse is not a valid response. This is to make sure that the NNAP is able to re-initialize without having to fall back to the long-term MUK. For example, if the CommissionResponse is lost during transfer and never reaches the NNAP, then the NNAP has two options: perform a new (fresh) re-initialization using the current CryptoContext (again) or try to do the same but with the new CryptoContext. If one fails, then the NNAP should try the other. If both options fail, then the NNAP may try again or eventually fall back to using the long-term MUK.

The LNAP should wait at most n seconds since the ChallengeRequest for the CommissionRequest. If n seconds passes without receiving a CommissionRequest, then the LNAP should abort the process. Deciding the value of n is left for the companion specification to decide as this is system-specific.

The LNAP should also limit the number of ChallengeRequest messages it will accept within a certain time period. For example, if an LNAP has received 100 challenge requests within one hour, then it should wait 1 h before allowing another 100 challenge requests. Care should be taken not to make it trivial for the network adversary to exploit this mechanism.

If an authentication failure occurs, then the NNAP should log an intrusion detection event that the head-end system can analyze and perform proper incident response. For example, the head-end system may decide to use the MUK to re-initialize the LNAP if the short-term keys do not work, however, doing so would cause an incident event to be triggered by the NNAP.

D.7.4 Security Suite Negotiation

The NNAP specifies the desired security suite (in the "Security Mode" field of the unencrypted FCM request header, see Table D.5). If the LNAP does not support the desired security mode, it will NACK the request and indicate in the response which security mode(s) it does support. The NNAP can then decide whether to proceed with one of the alternatives, or to report a security incompatibility. Care should be taken to avoid downgrade attacks as the NACKs can be forged by the network adversary. If possible, it is recommended to configure the NNAP to

only proceed with OSGP-AES-128-PSK, and to report a security event if the LNAP (attacker) responds with an unexpected list of supported security modes.

D.7.5 LNAP Commissioning

Besides exchanging security information, commissioning configures the LNAP for normal communication, and discovers capabilities of the LNAP. These pieces of information are combined in a single message transaction (CommissionRequest and CommissionResponse) to speed up commissioning.

The following fields of the commissioning messages are necessary for ISO/IEC 14908-1 addressing and message timing:

- domain_ID,
- subnet_ID,
- node_ID,
- non-group RX Timer.

The reader shall keep in mind that the given LNAP's domain_ID, subnet_ID, and node_ID is also tied to the current CryptoContext.

The following fields allow the LNAP's capabilities to be identified:

- Optional Features,
- Compatibility Set,
- Program ID,
- Inactive Phases,
- Hardware configuration Length,
- Hardware configuration record,
- LNAP Model (meter model).

D.7.6 Error Handling and Intrusion Detection

Error handling during the secure channel initialization is simple: any type of error state is fatal and forces the entire process to be aborted. Aborting the process also invalidates both transactions, which means none of the information transferred can be trusted. As a result, the NNAP has to initiate a *fresh* initialization *if* it decides to try again.

If an AuthenticationFailure is returned, then it should be logged and fed to an intrusion detection system for incident response. In practice, there should be no legitimate case of an AuthenticationFailure: either an attacker injected the error message or the key was replaced by something unexpected. Either way, this is an abnormal state and should be flagged.

D.7.7 Messages

D.7.7.1 Overview

Subclause D.7.7 specifies the four non-error messages used during secure channel initialization.

D.7.7.2 ChallengeRequest

Table D.3 shows the structure of challenge request to be sent by the NNAP.

Table D.3 – Challenge request

	Offset (byte)	Type	Name	A	E	Description
DC_ flags	byte 0	unsigned	ISO/IEC 14908-1 Code	Yes	No	0x06
	byte 1	unsigned	Configuration	Yes	No	Bitmap b0 – b2: FCM-Sub Code. 0 b3 – b5: Security mode. 2=AES-128 b6 – b7: reserved
	byte 2	unsigned	Optional Features	Yes	No	Bit 0: If set, NNAP supports MCK Bits 1-7: Reserved
	byte 3	octet-string(SIZE(16))	DC_rand	Yes	No	Random challenge generated by the NNAP
	byte 19	octet-string(SIZE(16))	DC_MAC	MAC	No	MAC covering the preceding data

D.7.7.3 ChallengeResponse

Table D.4 shows the structure of the LNAP response to the NNAP challenge request.

Table D.4 – Challenge response

	Offset (byte)	Type	Name	A	E	Description
M_ flags	byte 0	unsigned	ISO/IEC 14908-1 Code	Yes	No	0x06
	byte 1	unsigned	Upper layer Result Code	Yes	No	Success (0x00)
	byte 2	unsigned	Supported Security Modes	Yes	No	Bitmap of supported Security modes Bit 2 – AES-128 Bits 3-7 – Reserved
	byte 3	unsigned	Optional Features	Yes	No	Bit 0: If set, LNAP supports MCK Bits1-7: Reserved
	byte 4	octet-string(SIZE(16))	M_RAND	Yes	No	Random challenge value generated by the LNAP
	byte 20	octet-string(SIZE(20))	M_MAC	MAC	No	MAC covering the preceding data concatenated with the DC_RAND

D.7.7.4 CommissionRequest

Table D.5 shows the structure of the commission request sent by the NNAP.

Table D.5 – Commission request

	Offset (byte)	Type	Name	A	E	Description
FCM request header	byte 0	unsigned	ISO/IEC 14908-1 Code	Yes	No	0x06
	byte 1	unsigned	FCM-Sub Code	Yes	No	Bitmap b0 – b2: FCM-Sub Code. 1= Commission b3 – b5: Security Mode. 2= AES-128 b6 – b7: reserved
	byte 2	unsigned	Optional Features	Yes	No	A bitmap of optional features which may or may not be supported by the LNAP. The FCM response will indicate the complete set of supported optional features. This is reserved for manufacturer-defined options
FCM request payload	byte 3	unsigned	Non-group RX Timer	Yes	Yes	Bitmap: b0 – b3: Non-group RX Timer b4 – b7: reserved. Set to 0 The ISO/IEC 14908-1 non-group receive timer
	byte 4	octet-string(SIZE(3))	Domain ID	Yes	Yes	ISO/IEC 14908-1 domain ID. Domain ID will always be 3 bytes.
	byte 7	unsigned	subnet_ID	Yes	Yes	ISO/IEC 14908-1 subnet_ID
	byte 8	unsigned	node_ID	Yes	Yes	b0-b6: ISO/IEC 14908-1 node_ID
			Spare	Yes	Yes	b7: reserved
	byte 9	octet-string(SIZE(8))	Compatibility Set	Yes	Yes	Bitmap of compatibility settings
	byte 17	octet-string(SIZE(16))	MMK	Yes	Yes	
	byte 33	octet-string(SIZE(2))	URand	Yes	Yes	
	byte 35	octet-string(SIZE(16))	DMK	Yes	Yes	
	byte 51	double-long-unsigned	BSeq	Yes	Yes	
	byte 55	octet-string(SIZE(5))	Brand	Yes	Yes	

D.7.7.5 Commission Response

Table D.6 shows the structure of the LNAP commission response.

Table D.6 – Commission response structure

	Offset (byte)	Type	Name	A	E	Description
FCM response header	byte 0	unsigned	ISO/IEC 14908-1 Code	Yes	No	0x06
	byte 1	unsigned	Upper layer ACK	Yes	No	0x00 Success
	byte 2	unsigned	Supported Security Modes	Yes	No	Bitmap of supported Security modes
	byte 3	byte	reserved	Yes	Yes	Reserved
FCM response payload	byte 4	octet-string(SIZE(8))	Program ID	Yes	Yes	Meter Program ID
	byte 12	unsigned	Inactive Phases	Yes	Yes	LNAP inactive phases
	byte 13	unsigned	Reserved	Yes	Yes	Bitmap of active MEP ports (from System information)
	byte 14	unsigned	Test Point Flags	Yes	Yes	Test point status
	byte 15	unsigned	Hardware configuration Length	Yes	Yes	Length of Hardware configuration data
	byte 16	octet-string	Hardware configuration record	Yes	Yes	Hardware configuration data
	byte 17+n Length	octet-string	Meter Model #	Yes	Yes	LNAP model #

D.8 Secure Channel Communication

D.8.1 General

To communicate securely, a NNAP first needs to establish a shared, LNAP-specific CryptoContext as specified in Clause D.7. Once a CryptoContext has been established with the LNAP, and thereby the secure channel, a NNAP can start communicating securely with the LNAP, as illustrated in Figure D.7, showing an ordinary unicast request/response flow.

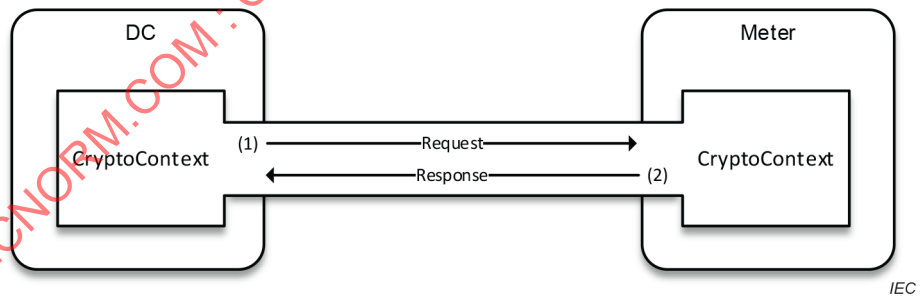

Figure D.7 – Crypto context establishment

Clause D.8 specifies how the NNAP and the LNAP use the shared CryptoContext to communicate securely using the underlying authenticated encryption mechanism.

The secure channel provides data confidentiality, authenticity, and replay protection for both unicast transactions and broadcast messages. Refer to Clause D.2 for a description of what unicast and broadcast communication is in this context.

D.8.2 The general process

D.8.2.1 Overview

Subclause D.8.2 outlines the general process of using the authenticated encryption mechanism for protecting and processing messages.

D.8.2.2 Protecting and Sending a Message

Figure D.8 illustrates how the authenticated encryption mechanism is used to protect a payload (plaintext header followed by a sensitive payload).

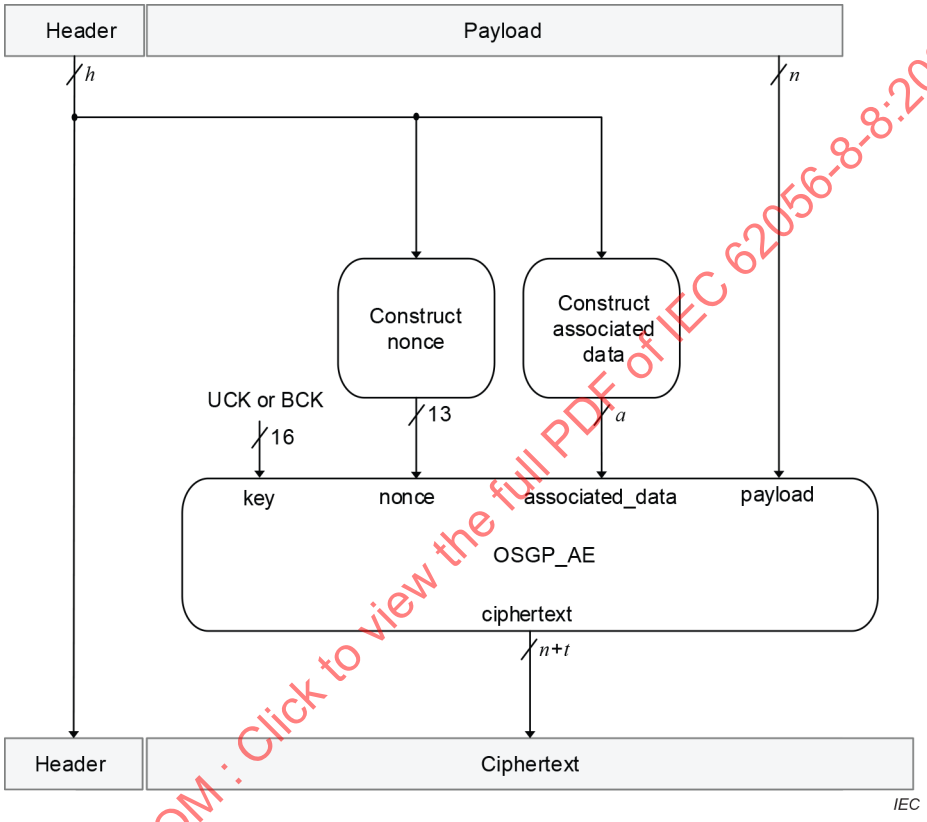

Figure D.8 – Authenticate encryption mechanism

The diagram below illustrates how the authenticated encryption mechanism is used to protect a message (plaintext header followed by a sensitive payload). In general, it works as follows:

a) **Construct nonce**: construct a nonce based on the header and contextual information;

b) **Construct associated data:** construct associated data based on the header and contextual information;

c) **Authenticated encryption:** use the OSGP_AE function to perform an authenticated encryption of the given payload which at the same time provides authenticity of the associated data (header and contextual information). The length of the output ciphertext is the sum of the length of the payload and the length of the MAC (t) as defined in D.4.3;

d) **Transmit:** send (plaintext) header followed by the ciphertext (the output of the OSGP_AE function).

D.8.2.3 Processing a Protected Message

Processing a protected payload (header || ciphertext) is shown in Figure D.9.

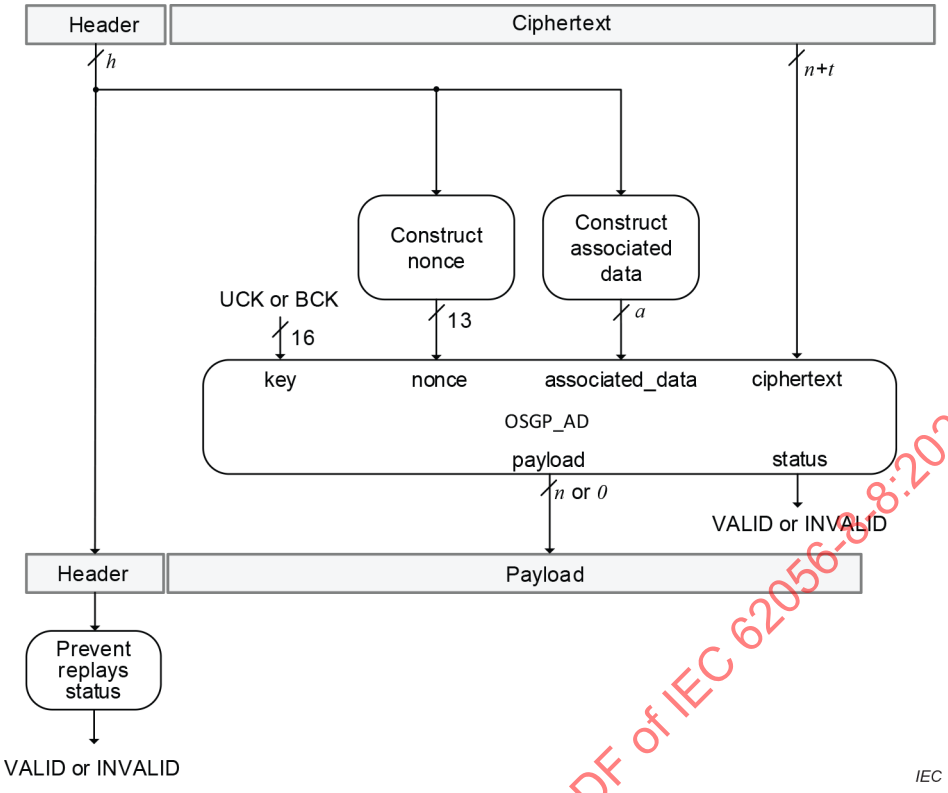

Figure D.9 – Authenticated decryption

Processing a protected PDU (header || ciphertext) is shown in the diagram above. In general, the process is as follows:

a) **Construct nonce**: re-construct the nonce based on the header and contextual information.

b) **Construct associated data:** re-construct associated data based on the header and contextual information.

c) **Authenticated decryption:** use the OSGP_AD function to perform an authenticated decryption of the ciphertext. If the authenticated decryption fails, then respond with AuthenticationFailure (D.12.2), and abort the process; otherwise, continue.

d) **Prevent replays:** use the (now authenticated) header information to check for replays. If the message is a replay, then respond with SequenceError (D.12.3) and abort the process; otherwise, continue.

e) **Process:** process authenticated payload.

D.8.3 Unicast communication

D.8.3.1 Overview

Securing unicast transactions using the secure channel follows the general process as described above.

D.8.3.2 Message construction

A NNAP request message consists of a header and a payload which is specified in Table D.7:

Table D.7 – Request message

	Offset (byte)	Type	Name	A	E	Description
Header	byte 0	unsigned	ISO/IEC 14908-1 Code	Yes	No	0x00
	byte 1	octet-string(SIZE(8))	USeq	Yes	No	Request sequence number
Payload	byte 5	unsigned	Upper layer Request Code	Yes	Yes	Upper layer command
	byte 6	octet-string	data	Yes	Yes	Command data

The USeq is incremented by one before constructing the request. Refer to the nonce creation clause for more information about the USeq.

A LNAP response message consists of a header and a payload and is specified in Table D.8:

Table D.8 – response message structure

	Offset	Type	Name	A	E	Description
Header	byte 0	unsigned	ISO/IEC 14908-1 Code	Yes	No	Set to 0x00
	byte 1	unsigned	Upper layer Result Code	Yes	No	Upper layer command result code
Payload	byte 2	octet-string	Upper layer response data	Yes	Yes	Upper layer result data

D.8.3.3 Unicast Nonce Construction

The nonce format is specified in Table D.9:

Table D.9 – Nonce construction

Offset (byte)	Type	Name	Part of Header	Description
0	unsigned	Flags	No	Bit fields set based on message type: Bit# 0 – Response (0 in request, 1 in response) Bit# 1 – FCM Message (0) Bit# 2-7 – Reserved (0)
1	octet-string(SIZE(4))	USeq	Yes	Request sequence number
5	octet-string(SIZE(6))	MID	No	Meter-unique identifier
11	octet-string(SIZE(2))	URand	No	Random data generated on a per-UCK basis

The USeq is an incrementing counter for each NNAP request transmitted under the same UCK. The NNAP increments the USeq by 1 (USeq = USeq + 1) before sending a request.

For the corresponding response, the LNAP uses the same value for the USeq, but changes the Response field in the Flags field from 0 to 1 when constructing the response nonce. Thus, as long as the NNAP makes sure to increment the USeq before sending a request, both the NNAP and the LNAP should be generating (key-)unique nonces as required by the underlying CCM algorithm.

USeq is stored in the CryptoContext shared between the NNAP and the LNAP (CryptoContext.USeq).

The URand is randomly generated by the NNAP on a per-UCK basis. The reason for including a random value (URand) within the nonce is twofold:

a) it provides an additional limited safety in terms of nonce-uniqueness, and

b) to make pre-computation attacks more difficult. The best way to address these types of attacks is to increase the key size/strength (e.g. 256-bit key vs. 128-bit key).

However, because of performance constraints, this is not an option. Including a random value in the nonce is therefore used instead to provide a limited amount of additional security against pre-computation attacks.

D.8.3.4 Unicast Associated Data Construction

The associated data cryptographically binds message-specific and channel-specific context to each message.

The associated data for requests and responses is specified in Table D.10:

Table D.10 – Associated Data construction

Offset (byte)	Type	Name	Description
0	octet-string(SIZE(13))	Nonce	The request or response nonce as specified in D.8.3.3.
13	octet-string(SIZE(3))	MDomainID	LNAP's ISO/IEC 14908-1 domain ID, i.e. CryptoContext.DomainID
16	unsigned	MNodeID	LNAP's ISO/IEC 14908-1 node_ID, i.e. CryptoContext.NodeID
17	unsigned	MSubnetID	LNAP's ISO/IEC 14908-1 subnet_ID, i.e. CryptoContext.SubnetID
18	octet-string	Header	For requests: this is the request header as defined in D.8.3.2, without the USeq (it's already in the Nonce field) For responses: this is the response header as defined in D.8.3.2

D.8.3.5 Replay Protection

A LNAP will only accept requests that contain an authentic USeq value that is greater than its current CryptoContext value.USeq (USeq > CryptoContext.USeq). After a successful message verification and replay check, the LNAP sets its CryptoContext.USeq to the USeq value in the request.

After a NNAP sends a request, it will only accept a response that was generated using the same USeq but with its Response bit set to 1 in the Flags. Since the USeq is part of the associated data for the response, this check is done during the authenticated decryption of the response.

D.8.3.6 Flow

The unicast transaction process is specified below.

Input:

- CryptoContext: the cryptographic context established between the NNAP and the LNAP after the last successful secure channel initialization as specified in Clause D.7.

Steps:

NNAP side

1) NNAP makes sure that CryptoContext.USeq < $2^{32} - 1$, if this is not the case, then it stops (it needs to renew the entire CryptoContext as specified in Clause D.7 in order to resolve this)
2) NNAP increments CryptoContext.USeq by one
3) NNAP constructs the request message as specified in D.8.3.2 using the (incremented) USeq from the CryptoContext (CryptoContext.USeq)
4) NNAP constructs a 13-byte request nonce as specified in D.8.3.3.
5) NNAP constructs the associated data as specified in D.8.3.4
6) NNAP performs an authenticated encryption of the request message and the associated data:
```
ciphertext = OSGP_AE(payload=payload,
                     associated_data=associated_data,
                     nonce=nonce,
                     key=CryptoContext.UCK)
```
7) NNAP sends header || ciphertext to LNAP

LNAP side

8) LNAP parses the request header according to the message format specified in D.8.3.2
9) LNAP constructs a 13-byte request nonce as specified in D.8.3.3, using the USeq from the header
10) LNAP constructs the associated data for the request as specified in D.8.2.3
11) LNAP performs an authenticated decryption of the request:
```
payload, status = OSGP_AD(ciphertext=ciphertext,
                          associated_data=associated_data,
                          nonce=nonce,
                          key=CryptoContext.UCK)
```
12) If status is INVALID, then LNAP responds with an AuthenticationFailure (D.12.2) and aborts the process; otherwise, the LNAP proceeds
13) If the USeq value in the (now authenticated) request header is less than or equal to CryptoContext.USeq, then LNAP responds with a SequenceError (D.12.3) and aborts the process; otherwise the LNAP proceeds
14) LNAP sets CryptoContext.USeq = USeq
15) LNAP processes the authenticated payload
16) LNAP assembles a response message according to the message format as specified in D.8.3.2 using the USeq from the CryptoContext
17) LNAP constructs a 13-byte response nonce as specified in D.8.3.3 making sure the Response bit is set in the Flags field of the nonce
18) LNAP constructs the associated data for the response as specified in D.8.2.3
19) LNAP performs an authenticated encryption of the response message
```
ciphertext = OSGP_AE(payload=payload,
                     associated_data=associated_data,
                     nonce=nonce,
                     key=CryptoContext.UCK)
```
20) LNAP sends header || ciphertext to NNAP (protected response message)

NNAP side

21) NNAP parses the response header as specified in D.8.2.1 .
22) NNAP constructs a 13-byte response nonce as specified in D.8.3.3 using the USeq from the CryptoContext (CryptoContext.USeq)
23) NNAP constructs associated data for the response message as specified in D.8.2.3.

24) NNAP performs an authenticated decryption of the response:
status, payload = OSGP_AD(ciphertext=ciphertext,
associated_data=associated_data,
nonce=nonce,
key=CryptoContext.UCK)

25) If status is INVALID, then NNAP should log the event and abort the process; otherwise, NNAP proceeds

26) NNAP processes the authenticated payload

D.8.3.7 Error Handling and Intrusion Detection

Ultimately, it is up to the project companion specification to decide how to handle security-related errors during unicast communication. The following are only helpful points and recommendations for implementers, not requirements.

Assuming the NNAP is implemented as per this specification, a SequenceError should only mean two things in practice:

a) that the UCK was compromised, and the adversary has incremented the USeq during unicast transactions, leaving the NNAP's USeq in the past, or,

b) an adversary injected a SequenceError during a unicast transaction, which means there is an active adversary on the domain.

Likewise, if the NNAP receives an AuthenticationFailure, it should only mean that the CryptoContext was changed by someone other than the NNAP.

Keep in mind that the underlying transport protocol performs its own data integrity check of packets using a CRC, which should, in practice, mean that an AuthenticationFailure is not caused by a noisy link, but by a malicious actor.

To summarize, in practice, there should be no legitimate cases in which these security-related error messages can occur. Thus, observing any of these types of errors should be fed into an intrusion detection system for incidence response. It is highly recommended to log and act on these security-related errors appropriately, since it may very well indicate a LNAP compromise or the presence of an active network adversary.

D.8.4 Broadcast Communication

D.8.4.1 Overview

Broadcasts are one-way messages sent by the NNAP to a group of LNAPs within the given domain. The receiving LNAPs do not reply to broadcasts. Broadcast messages are secured using the broadcast-specific key (BCK). A NNAP can update its current BCK using a separate mechanism as specified in D.10.2.

D.8.4.2 Message Construction

The broadcast message format is specified in Table D.11.

Table D.11 – Broadcast message

	Offset (byte)	Type	Name	A	E	Description
Header	byte 0	unsigned	ISO/IEC 14908-1 Code	Yes	No	0x00
	byte 1	octet-string(SIZE(4))	BSeq	Yes	No	Broadcast sequence number
Payload	byte 5	unsigned	Upper layer Request Code	Yes	Yes	Upper layer command
	byte 6	octet-string	Upper layer request data	Yes	Yes	Upper layer command data

The BSeq is a monotonically incrementing counter for each NNAP broadcast message transmitted under the same BCK. The NNAP increments the BSeq by 1 before sending a broadcast message.

D.8.4.3 Broadcast Nonce Construction

The nonce format is specified in the Table D.12:

Table D.12 – Broadcast nonce

Offset (byte)	Type	Name	Also in Header	Description
byte 0	unsigned	Flags	No	Bit fields set based on message type (0x00)
byte 1	octet-string(SIZE(4))	BSeq	Yes	Broadcast message sequence number
byte 5	octet-string(SIZE(3))	DomainID	No	NNAP's Domain ID
byte 8	octet-string(SIZE(5))	BRand	No	Random bytes

The BSeq is incremented by one before constructing the nonce. See D.8.4.2 for more information.

The BRand is generated on a per-BCK basis. See D.8.3.3 for the rationale of including random bytes in the nonce.

D.8.4.4 Broadcast Associated Data Construction

The associated data for broadcast messages is specified in Table D.13:

Table D.13 – Broadcast Associated Data

Offset (byte)	Type	Name	Description
0	octet-string(SIZE(13))	Nonce	The broadcast nonce as specified above in D.8.4.3.
13	octet-string	Header	Broadcast header as specified in D.8.4.2 without the BSeq (it's already in the nonce field).

D.8.4.5 Replay Prevention

A LNAP will only accept broadcast messages that contain an authentic BSeq value that is greater than its current CryptoContext.BSeq (header.BSeq > CryptoContext.BSeq). After a successful message verification and replay check, the LNAP sets its CryptoContext.BSeq to the BSeq value found in the broadcast message header.

D.8.4.6 Flow

The broadcast process is specified below.

Input:

a) CryptoContext: the current cryptographic context established between the NNAP and the domain (i.e. the BSeq, BRand, and the BCK in the current CryptoContext).

Steps:

NNAP side

1) NNAP makes sure that CryptoContext.BSeq < $2^{32} - 1$, if this is not the case, then it stops (it needs to renew the BCK and the corresponding BSeq as specified in D.10.2 in order to resolve this);
2) NNAP increments CryptoContext.BSeq by one (CryptoContext.BSeq += 1);
3) NNAP constructs a 13-byte broadcast nonce as specified in D.8.3.2;
4) NNAP constructs the associated data as specified in D.8.4.4. NNAP performs an authenticated encryption of the broadcast message and the associated data:

```
ciphertext = OSGP_AE(payload = payload,
    associated_data=associated_data
    nonce=nonce,
    key=CryptoContext.BCK)
```

5) NNAP sends header || ciphertext to LNAP.

LNAP side

6) LNAP parses the broadcast message according to the format specified in D.8.4.2;
7) LNAP constructs a 13-byte broadcast nonce as specified in D.8.4.3;
8) LNAP constructs the associated data for the request as specified in D.8.4.4;
9) LNAP performs an authenticated decryption of the request:
10) payload, status, = OSGP_AD(ciphertext=ciphertext,
    associated_data=associated_data,
    nonce=nonce,
    key=CryptoContext.BCK)
11) If status is INVALID, aborts the process; otherwise, LNAP proceeds
12) If the BSeq in the now authenticated broadcast message header is less than or equal to CryptoContext.BSeq, then LNAP aborts the process; otherwise, LNAP proceeds
13) LNAP sets CryptoContext.BSeq = BSeq (the BSeq is from in the authenticated header)
14) LNAP processes the authenticated broadcast message.

D.9 Firmware Downloading

The download process shall allocate a range of sequence numbers where the range is greater than the total number of packets required to download the firmware image. The sequence number in an individual packet will be the starting sequence number + <packet number>. If it is necessary to re-send a given packet, the same sequence number shall be used, and also the same Procedure Sequence Number. This is allowed, because the contents of the packet will be identical in both cases, and thus the re-send is equivalent to a layer 2 retry.

For OSGP-AES-128-PSK devices, the switchover unicast request procedure includes an additional firmware image MAC. This MAC is computed over the entire downloaded image using the meter-unique MFWK as follows:

OSGP_MAC(data=firmware image, key=MFWK)

Prior to switching into the new image, the LNAP shall verify the MAC using OSGP_MAC_VERIFY, and shall reject the image if there is a mismatch. The same authentication shall also be used for downloading image for devices which are connected to the MEP. In this case, the LNAP shall only validate the image if it is able to store the entire image in a non volatile memory. If the image size exceeds those limits, then it shall fail verification, but the MEP will still be notified when the new image is complete.

A new firmware download issue, introduced with AES support, is the interaction between the Broadcast Resync function and download. If the BCK were changed during a firmware download, LNAPs would reject the firmware image packets until their BCK was updated.

In order to avoid this situation, the head-end system shall make sure to disable updates to the BCK while a firmware download is in progress. Additionally, if the BCK is about to expire, the head-end system will initiate the BCK change and allow that to be completed prior to initiating the firmware download. Also, the NNAP shall skip any devices that have a pending BCK update during the firmware download.

D.10 Key Management

D.10.1 Renewing the Short-term, Meter-unique Keys

The short-term, meter-unique keys are updated by re-initializing a new secure channel as specified in Clause D.7.

D.10.2 Renewing the Short-term, Domain-unique Keys

Changing the short-term, domain-wide keys can also be done by re-initializing a new secure channel with a new DMK value.

To update the short-term, domain-wide keys separately from the secure-channel initialization, the NNAP shall perform the following steps.

1) NNAP generates a new DMK:
   DMK = OSGP_CSPRNG(16)
2) NNAP generates a new BRand:
   BRand = OSGP_CSPRNG(5)
3) For each LNAP, CurrentLNAP, in the NNAP's domain:
   a) NNAP assembles UpdateDomainKeysRequest (specified below)
   b) NNAP protects and sends the UpdateDomainKeysRequest using the secure unicast communication process as specified in D.8.3.
   c) CurrentLNAP processes the received request using the secure unicast communication process as specified in D.8.3.
   d) If the received request is valid, then CurrentLNAP updates its CryptoContext:
      1. CryptoContext.BCK = OSGP_KDF(
         input_key=UpdateDomainKeysRequest.DMK,
         context=CryptoContext.DomainID || 0x000000,
         label="bck")
      2. CryptoContext.LTK = OSGP_KDF(
         input_key=UpdateDomainKeysRequest.DMK,
         context=CryptoContext.DomainID || 0x000000,
         label="ltk")[:12]
      3. CryptoContext.BRand = UpdateDomainKeysRequest.BRand
      4. CryptoContext.BSeq = UpdatedomainKeysRequest.BSeq
   e) CurrentLNAP assembles a UpdateDomainKeysResponse (specified below)

f) CurrentLNAP protects and sends the UpdateDomainKeysResponse using the secure communication process as specified in D.8.3.

g) NNAP processes the received response using the secure communication process as specified in D.8.3.

D.10.3 UpdateDomainKeysRequest

The UpdateDomainKeysRequest is specified in Table D.14:

Table D.14 – Update Domain Keys request

	Offset (byte)	Type	Name	A	E	Description
Header	byte 0	unsigned	ISO/IEC 14908-1 Code	Yes	No	0x06
	byte 1	unsigned	FCM-Command	Yes	No	Bitmap b0 – b2: FCM-Command. 2 – Set Broadcast key b3 – b5: Security mode b6 – b7: reserved
	byte 2	unsigned	Optional Features	Yes	No	A bitmap of optional features which may or may not be supported by the LNAP.
	byte 3	octet-string(SIZE(4))	Sequence Num	Yes	No	The meter's unicast sequence number.
Payload	byte 7	octet-string(SIZE(16))	DMK	Yes	Yes	See Clause D.6
	byte 23	octet-string(SIZE(5))	BRand	Yes	Yes	Random data to be used in NNAP-LNAP broadcast messages
	byte 28	double-long-unsigned	BSeq	Yes	Yes	Starting sequence number for NNAP-LNAP broadcast messages

D.10.4 UpdateDomainKeysResponse

The UpdateDomainKeysResponse is specified in Table D.15:

Table D.15 – Update Domain Keys response

	Offset (byte)	Type	Name	A	E	Description
Header	byte 0	unsigned	ISO/IEC 14908-1 Code	Yes	No	0x06
	byte 1	unsigned	Upper layer ACK	Yes	No	0x00 Success
	byte 2	unsigned	Supported Security Modes	Yes	No	Bitmap of supported Security modes
	byte 3	unsigned	Optional Features	Yes	No	A bitmap of optional features supported by the LNAP.

D.11 Updating the Long-term, Meter-unique Keys

D.11.1 General

Updating the MUK is outside the scope of this specification.

D.11.2 Key Validity Periods

It is ultimately up to the companion specification to define an appropriate key renewal policy for the specific system at hand. This security suite merely provides secure means of doing so and general recommendations. Refer to Clause D.13 for the latter.

The reader has to keep in mind that managing the MUK is outside the scope of this security suite.

D.12 Error Messages

D.12.1 Overview

This clause defines the security-related LNAP error responses used during the secure channel initialization (see Clause D.7) and secure unicast communication (see D.8.3).

D.12.2 AuthenticationFailure

If the LNAP is unable to authenticate a message, then it will send an authentication failure response as specified in Table D.16:

Table D.16 – AuthenticationFailure message

Offset (byte)	Type	Name	Description
0	unsigned	ISO/IEC 14908-1 Code	Code of the request message: 06
1	unsigned	NACK code	0x0B Authentication Failure
2	unsigned	Supported Security Modes	Bitmap of supported Security modes
3	unsigned	Optional Features	A bitmap of optional features supported by the LNAP.

D.12.3 SequenceError

If the sequence number specified in a unicast request is not valid, then the LNAP will reject the request and send an invalid sequence number response as specified in the Table D.17:

Table D.17 – SequenceError message

Offset (byte)	Type	Name	Description
0	unsigned	ISO/IEC 14908-1 Code	Code of the request message: 06
1	unsigned	NACK code	0x0C Sequence number failure
2	unsigned	Supported Security Modes	Bitmap of supported Security modes
3	unsigned	Optional Features	A bitmap of optional features supported by the LNAP.

D.13 Security considerations

D.13.1 Reasoning

D.13.1.1 Choosing the Cryptographic Primitives

D.13.1.1.1 General

Subclause D.13.1.1 provides the reasoning for the choice of cryptographic primitives used in this security suite.

The candidates for the cryptographic primitives were selected primarily with respect to the possible scarce resources of the target device. As these devices could be up to 15 years old, public-key cryptography (even elliptic curve cryptography (ECC) and simple Diffie-Hellman key agreements) would impose too much of a cryptographic overhead; both in terms of computation, code space, number of packets needed, and packet payload size.

Because of this substantial limitation, a symmetric (secret-key) based cryptographic system was selected requiring only one main primitive to be implemented: a secure symmetric encryption block cipher. The cipher is re-used in different applications/modes of operation to provide the needed cryptographic primitives to support secure communication, while still catering for the aforementioned limitations.

The remainder of D.13.1.1 provides reasoning for the choice of block cipher, message authentication construction, and authenticated encryption construction.

D.13.1.1.2 Encryption Block Cipher AES-128

The security guarantees of both CMAC and CCM critically depends on the security assurance provided by the underlying block cipher.

AES-128 was chosen amongst two other candidates: AES-256 and ChaCha20 (see IETF RFC 7539).

AES-128 was chosen over AES-256 because of the limited available device resources that this security suite is designed for; AES-256 does not provide an acceptable computational performance/security balance. In contrast, AES-128 provides a more-than-sufficient security margin for the foreseeable future while at the same time achieving an acceptable performance overhead. In addition, latest cryptoanalysis of AES-256 shows weaknesses not found in the AES-128 variant. Although for AES-128, the latest attack is mostly applicable if related-keys are used, attacks will only get better and we would rather choose a block cipher algorithm that shows the least structural weaknesses.

AES-128 was chosen over ChaCha20 because it is, as opposed to ChaCha20, more widely approved by the security industry and standards organizations such as NIST and IEEE. AES-128 remains the safest choice for this security suite when it comes to wide approval.

D.13.1.1.3 Message Authentication CMAC

A message authentication function is needed during the handshake transaction of a secure channel (re-)initialization to provide message authenticity (not confidentiality). The security of the secure channel relies on the security assurances provided by the message authentication function making it a critical component in this security suite.

Because of the design goal of using as little code space as possible, a re-use of the already available AES-128 block cipher for message authentication function is preferred. As a result, hash-based message authentication algorithms (e.g. HMAC-SHA-256) are not considered since they would require the availability of a new cryptographic hash function, such as SHA-256.

The CMAC mode of authentication was chosen among two other candidates: CBC-MAC and GMAC (GMAC is essentially an implementation of GCM, which restricts its input to the non-confidential data input, i.e. the associated data input).

CMAC was chosen over CBC-MAC because of CBC-MAC's security deficiencies as a stand-alone MAC function. Although CBC-MAC is mostly known for its weakness in a variable-length application, the fact that it is difficult to get right was the main reason for choosing CMAC over CBC-MAC.

Refer to D.13.1.1.4 for the reasoning of choosing CMAC over GMAC.

D.13.1.1.4 Authenticated Encryption with Associated Data (AEAD): CCM

An authenticated encryption with associated data primitive is used during secure communication. This means that the security of the majority of the data transferred on the network relies on the security assurances provided by this mode, and its underlying block cipher.

CCM was chosen amongst two other candidates: GCM and OCB.

CCM was chosen over OCB because of OCB's licence, which restricts its application.

There are several reasons why CCM was chosen over GCM:

a) CCM has been used for several years and is still widely being used to successfully protect network communications in embedded distributed systems, such as wireless networks (WPA2). These systems share similar performance requirements with this security suite.

b) CCM is significantly simpler to implement securely in software, and the targeted hardware platform, for running this security suite on, is not assumed to have hardware (or otherwise built-in) support for a vetted and secure GCM implementation. In cases where it is not possible to simply use a secure software implementation of GCM, implementers need to be extra careful in how they implement GCM compared to the simpler CCM.

c) Although GCM is theoretically faster than CCM, the code space requirement for GCM is significantly more than CCM. In this security suite, code space is critical and the performance of CCM was deemed acceptable during initial testing.

d) Because of the bandwidth-constrained target network that this security suite is designed for, it is necessary to truncate the MAC during secure communication in order to achieve acceptable availability of the system. Additionally, there are currently no widely used and secure protocols that use a GCM MAC lengths below 96 bits. In contrast, using a 64-bit MAC with CCM is used in several protocols including CCMP-128 used in IEEE Std 802.11ac-2013, a widely used and analyzed protocol.

D.13.1.2 The design of the Secure Channel Initialization Flow

The secure channel initialization shall be able to securely establish a shared CryptoContext.

The decision of combining LNAP commissioning and secure channel initialization was made to lower the number of transactions needed. That way, establishing a secure channel only requires one extra transaction (the handshake), as opposed to two, and requires minimal effort to support, since LNAP commissioning is already there. In addition, logic in OSGP is already in place to support LNAP commissioning, and adding a few new fields to the LNAP commissioning process (FCM) imposes a small impact.

Requiring that the random numbers in the handshake transaction (DC_rand and M_rand) are freshly generated for each handshake achieves two critical goals: it prevent replays (freshness) and it provides the following commissioning transaction with a nonce it can use for performing an authenticated encryption/decryption of the CommissionRequest and the corresponding CommissionResponse.

The M_mac is calculated over a set of LNAP flags, both LNAP and NNAP random numbers (M_rand and DC_rand), and the meter-unique identifier (MID). The M_mac is used as part of the associated data input when processing the following commission transaction. The M_mac therefore cryptographically binds the handshake transactions with the commission transaction. As a result, if the authenticated decryption of the CommissionRequest succeeds, then mutual authentication has been achieved and the contents of the request is to be trusted to come from the NNAP.

This design achieves mutual authentication and key distribution in one step thereby saving one transaction. This is a significant optimization in real-world systems.

In addition, the nonce used in the commission messages consists of parts of both the DC_rand and the M_rand from the handshake, which further binds the freshness of the handshake transaction to the commission transaction.

D.13.1.3 Choosing the Length of the CCM MAC

The cryptographic payload overhead is defined by the length of the MAC output of CCM (T_{len}) in SP800-38C, which is used for every unicast and broadcast message in this security suite.

Because of the low propagation speed, narrow bandwidth, and high data corruption probability of the target system's network, the cryptographic payload overhead shall be minimal. Current deployments have reached acceptable reliability and performance by limiting the cryptographic overhead to 64 bits. This is the primary reason for choosing a 64-bit MAC.

One of the ways to address the security concerns with a truncated MAC is to shorten the validity period of the key used (i.e. renew the key more often – see Clause D.10). Refer to D.13.2 for key validity period recommendations that explicitly takes the truncated MAC into account.

D.13.2 Recommendation and Guidance for Implementers

D.13.2.1 Overview

This subclause provides recommendations and guidance for users of features described in this annex.

D.13.2.2 Validity Period for Short-term Keys

A key's validity period in this security suite should be defined by a time period and a key use count. The key should be updated as specified in Clause D.10 if its time period has ended or if it has reached its maximum use count, whichever comes first. That way, all keys will be updated regularly, and the risk of inauthentic messages passing the authenticated decryption process can be controlled.

The time period for a key should be set to the lowest possible period that does not decrease the availability of the network to an unacceptable level, because of the disruptiveness of the key renewal process. Since this is a very system-specific choice, this specification does not make any concrete recommendation.

As for key use count, the recommendation is to follow the guidance provided by NIST Special Publication SP800-38C, paying special attention to Appendix B. In Appendix B, two relevant variables are defined:

$Risk$ The highest acceptable probability for an inauthentic message to pass the decryption-verification process.

$MaxErr$ The maximum number of times that the output of any implementation of the decryption-verification process can be INVALID before the key is retired.

In this case, $Risk$ is the highest acceptable probability for a LNAP to accept an inauthentic unicast/broadcast message. $MaxErr$ is the number of times a LNAP rejects an inauthentic unicast message and, as a result, sends back an AuthenticationFailure message.

This means that any LNAP on the network can be used to check if a given message is authentic or inauthentic under the LNAP's current UCK (i.e. the LNAP is a message verification oracle). As a result, making sure that the $Risk$ is acceptable is crucial, otherwise, the security suite would become vulnerable to message forgery attacks.

In order to determine the appropriate value for $MaxErr$ based on a given $Risk$, SP800-38C provides the following inequality that shall be satisfied:

$$T_{\text{len}} \geq \lg\left(\frac{MaxErr}{Risk}\right)$$

Where lg(x) returns the base 2 logarithm of x and T_{len} is the length of the CCM MAC in bits. In this case, with a fixed MAC length of 64 bits, the inequality that shall be satisfied is:

$$64 \geq \lg\left(\frac{MaxErr}{Risk}\right)$$

One should choose an acceptable $Risk$ and then adjust the $MaxErr$ accordingly to satisfy the inequality.

However, keep in mind that a LNAP should not accept any more requests if $MaxErr$ has been reached, as it needs to have its key renewed. This creates an opportunity for the network adversary to force a given LNAP to use up its key by continuously sending inauthentic messages to the LNAP. Thus, when bounding the $MaxErr$, it is important to know how fast an adversary is able to query the LNAP (i.e. sending a request and receiving a response), as this makes it possible to estimate the time it takes for an adversary to force the retirement of a given LNAP's UCK.

Table D.18 provides three examples of varying risks but with fixed network performance.

Table D.18 – Example of risks

MaxErr	*Risk*	Trials/s	Time to retire 1 UCK by an active adversary in days
1,048,576	2^{-44}	2	6
4,194,304	2^{-42}	2	24
8,388,608	2^{-41}	2	49

Where trials/s is the number of unicast transactions that can be performed per second.

Again, choosing $MaxErr$ and $Risk$ should be based on the companion specification own assessment of the underlying system. This specification does not make any concrete recommendations, only guidance and examples.

Annex E
(normative)

Repeating mechanism

E.1 Overview

This Annex describes the mechanism used by the Repeating entity of the Adaptation layer for repeating packets over the ISO/IEC 14908-3:2012 PLC network. PLC networks are susceptible to noise and attenuation, making it difficult for devices to hear each other over distance. In addition, the nature of the PLC network can cause an ISO/IEC 14908-1:2012 router to hear the same packet from both sides of the topological path, making subnet-based routing unreliable.

To overcome these problems, the Adaptation layer uses for downlink communication a repeating mechanism in which each packet specifies the path of devices, between the sender and the target, which can hear each other well. All LNAPs (metering devices) inside the network have repeating capability and may be repeaters. Each LNAP, acting as a repeater along the path of the packet, passes the packet to the next LNAP on the path. The list of repeating LNAPs is specified by the originator of the transaction. The NNAP (Data Concentrator -DC-) is the only originator allowed, and maintains the repeating topology of the network and knowledge about the optimal path to each LNAP. Since each message contains the repeating path, and LNAPs never originate transactions, it is not necessary for LNAPs to have knowledge about its neighbours. For uplink communication, the packet uses in the reverse order the path already used by downlink communication.

When the NNAP wants to send a message to an LNAP that it cannot reach directly, it sends the message through other LNAPs that can reach the final LNAP. These repeating LNAPs have specific roles (see Clause E.2). There are also considerations regarding IEC 14908 Transport layer service types, timing, and address types that shall be handled. See E.3.2, E.3.3 and E.3.4.

E.2 Terms and definitions

E.2.1
Proxy Target
node with which the Proxy Source communicates, either directly or via Proxy Repeaters

Note 1 to entry: The Proxy Target is typically the Metering Device as specified in Figure 1.

E.2.2
Proxy Source
originator of the IEC 14908-1 transaction, and it is always the NNAP (DC)

Note 1 to entry: The Proxy Source has knowledge of the devices topology of the network, and maintains a table containing for each device the unicast address (Unique_Node_ID, and Node_ID), the list of Proxy Repeaters and Proxy Agent to use for reaching the related Proxy Target.

E.2.3
Proxy Agent
LNAP that communicates directly with the Proxy Target

Note 1 to entry: The Proxy Agent forwards the AdpPDU directly to the target. A Proxy Agent is a LNAP without application function embedded (see Figure 1).

E.2.4
Proxy Repeater
LNAP repeater located between the Proxy Source and the Proxy Agent

Note 1 to entry: A Proxy Repeater is an LNAP performing the repeating functionality at a given point of time.

E.3 Protocol specification

E.3.1 Overview

In the downlink direction, each Proxy Repeater along the path has the responsibility to store during the lifetime of the transaction – see E.3.8 – , and forward in both directions, the data received to the next Proxy Repeater or, at the end of the path, the Proxy Agent. In the uplink direction, the Proxy Repeaters forward data back to the Proxy Source.

E.3.2 Addressing

The scope of the repetition path specification is limited to a specific domain managed by the NNAP. The Proxy Repeaters and Proxy Agents shall use the same Domain_ID as the Proxy Source. For performing this, they shall be already configured in the domain of the Proxy Source during commissioning. See Clause F.3, Item f).

The addressing modes between the Proxy Agent and Proxy Target can also be any of the existing addressing modes (subnet/node, Unique node_Id, broadcast). See ISO/IEC 14908-1:2012, 10.3.

E.3.3 Service types

The Proxy Source can specify any of the IEC 14908-1 Transport layer service types (unackd, unackd-rpt, ackd, request/response – see ISO/IEC 14908-1:2012, 10.4) for the Proxy Target.

If the target service type is unackd, then the message shall be sent as unackd throughout the repeating.

For all the other service types, the message shall be sent as request/response until it reaches the Proxy Agent. This ensures that the message is reliably propagated to the Proxy Agent. The Proxy Agent then shall use the originally intended target service type to send the message to the Proxy Target. See E.4.2.4.2.

E.3.4 Timers

The Proxy Agent uses conventional IEC 14908-1 Transport layer timer and retry values to communicate with the Proxy Target. See ISO/IEC 14908-1:2012, B.4.11.

The Proxy Source and Proxy Repeater(s) use a set of control values different from that of the Proxy Agent. See E.4.2.3.

The Proxy Source provides the Proxy Repeaters with the number of retries to use. The retry timer value is also uniform across the Proxy Repeaters.

The maximum ISO/IEC 14908-1 transaction timer is 3 s. This may not be long enough to accommodate the worst case number of hops, particularly if authentication – see E.3.7 – is used. The value should be longer than the entire transaction time of the Proxy Agent/ Proxy Target transaction plus the round trip from the Proxy Agent back to the Proxy Source. For example, if the Proxy Agent/ Proxy Target transaction used 3 retries spaced at 192 milliseconds and there is one Proxy Repeater, then the Proxy Source/ Proxy Repeater should wait at least 1 s before retrying. Therefore, Proxy Source and Proxy Repeaters will use a special Long timer and specify it in the ProxyHeader.

The Proxy Repeaters and Proxy Agents will need to have ISO/IEC 14908-1 non-group receive timers long enough to accommodate the elongated timers and their retry counts. This will still be constrained to a total of 24 s, the limit specified in IEC 14908-1. See ISO/IEC 14908-1:2012, 9.1 for more information.

E.3.5 Request flow

The Proxy Source sends the AdpPDU by forming an ISO/IEC 14908-1 packet with proxy parameters. The proxy parameters include, among other information, the proxy parameters, the list of the Proxy repeaters and Proxy Agent.

When the Proxy Repeater receives the downlink packet, it processes the proxy parameters by stripping off the next repeater or agent address and forwarding the message to the next Proxy Repeater or to the Proxy Agent in the case when it is the last Proxy Repeater. The number of Proxy Repeaters is limited to 6.

The Proxy Agent strips off the Proxy parameters before relaying AdpPDU directly to the Proxy Target. It also sets the service type to be the type specified by the Proxy Source for the Proxy Target.

The proxied messages arrive at the Proxy Target as if they were originated directly from the Proxy Source except that the source address of the message will be that of the Proxy Agent.

E.3.6 Response flow

Because each Proxy Repeater establishes a transaction with the downstream node, the upstream response is sent using the standard ISO/IEC 14908-1 request/response mechanism (always subnet/node addressed). This means that for request/response transactions, while the response AdpPDU relayed upstream will be identical to that originally sent by the Proxy Target, the source address of the response received by the Proxy Source from the first Proxy Repeater will be that of the Proxy Repeater, not that of the Proxy Target.

It is the responsibility of each Proxy Repeater/ Proxy Agent to buffer any response it receives. This way, if a response is lost in transmission, and a request retry is received from the Adaptation layer of the previous Proxy Repeater – see Figure E.4 –, it will not be necessary to re-fetch the response from the downstream path – see Figure E.3 and Figure E.4. As the transaction layer of the ISO/IEC 14908 protocol stack does not buffer responses that contain data, this buffering shall be done by the Adaptation layer.

In the case that the Proxy Agent/ Proxy Target transaction is not a request/response but the Proxy Repeater/ Proxy Agent transaction is, the response will be a simple one-byte ISO/IEC 14908 completion code conveying success or failure. For example, if the Proxy Agent/ Proxy Target transaction is unackd-rpt broadcast, then upon sending the last repeat packet, the Proxy Agent would send a "success" response upstream. Similarly, if the Proxy Agent/ Proxy Target transaction is request/response and no response is received, a "failure" response will be sent uplink along with the subnet/node address of the node detecting the failure.

The Proxy Source would map these special codes into ISO/IEC 14908-1 completion events of the target – see Table E.8 and Table E.9.

E.3.7 Authentication

IEC 14908 authentication – see ISO/IEC 14908-1:2012, 11.11 – is used for network management messages to protect against unauthorized changes of the ISO/IEC 14908 configuration.

When the Proxy Source sends the ISO/IEC 14908-1 authenticated messages to the Proxy Target, then authenticated messages shall be used by all nodes along the path. Each hop is authenticated separately. For example, the first Proxy Repeater will challenge the Proxy Source and receive a successful reply, before relaying the proxied message on to the next Proxy Repeater or Proxy Agent.

In order to support changing of authentication keys, it is also necessary for the Proxy message to be able to convey an authentication key – see Figure E.9 –, for the Proxy Agent to use when communicating with the Proxy Target. This covers the case where the Proxy Agent and Proxy Target have different authentication keys. There is no provision to cover the case where the Proxy Source, Proxy Repeater and Proxy Agent have different keys. The Proxy Source shall ensure that the keys are the same in the intermediate points of the proxy chain before using authentication. The Proxy Agent shall use the key only for that outgoing transaction.

The authentication key passed – see E.4.2.4.3 – down for such messages will be in the form of a key increment (see ISO/IEC 14908-1:2012, 11.15). The Proxy Agent will use its own key, and byte-wise add the key increment to it, to derive the Proxy Target's key. This avoids sending the key in the clear.

E.3.8 Examples

The following figures illustrate exchanges between a Proxy Source and a Proxy Target with one Proxy Repeater.

Unacknowledged message:

Figure E.1 shows the flow from a Proxy Source to the Proxy Target for an unacknowledged message.

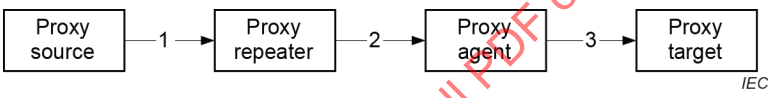

Figure E.1 – Unacknowledged message exchange

Request/Response message:

Figure E.2 shows the flow from a Proxy Source to the Proxy Target for a request/response message. There are 6 steps needed for the whole transaction completion. Transmission errors may occur at any step, as illustrated in Figure E.3 and Figure E.4.

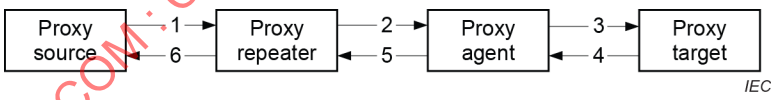

Figure E.2 – Request / Response message

In Figure E.3, the error occurs during the request transmission from the Proxy Agent to the Proxy Target. By having no response from the Proxy Target after the timer elapsed (step 3), the Proxy Agent which has stored the request sent to the Proxy Target until it gets the acknowledgement back, retries its transmission (Step 4). This last one was successful. The reader has to pay attention to the fact that the ProxyTxCtrl field of the proxy parameters holds the timer to be used by the Proxy Repeaters and the number of retries as well. The Proxy Source and Proxy Repeaters do not retry yet because their Retry Timer is longer than the timer at the Proxy Agent.

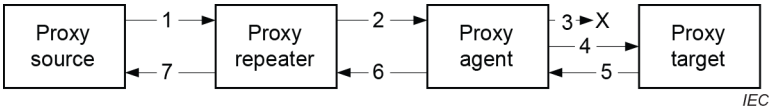

Figure E.3 – Error during request transmission

In Figure E.4, the transmission error occurs when the Proxy Agent is sending the response back to the Proxy Repeater. This transmission error has been detected by the Proxy Repeater when its own retry timer expires. Then the Proxy Repeater, which has stored the request sent to the Proxy Agent, until it gets the acknowledgement back, retries the transmission of step (7), the request already performed in step (2). As the Proxy Agent stored the response received from the Proxy Target, it only has to provide it without any query to the Proxy Target. Note that the Proxy Source also timed out and sent step (6), but this has no consequence.

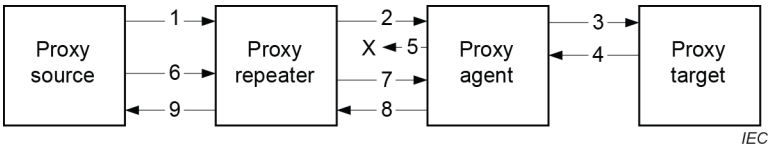

Figure E.4 – Error during response transmission

E.3.9 Broadcast

For broadcast messages, the service type from Proxy Source to Proxy Repeaters and Proxy Agent shall be request/response or unackd. The Proxy Agent again has the responsibility to change the service type to the one originally intended for the target, as specified in the ProxyTrailer.

Unacknowledged broadcasts are handled in a special way. In this case, each Proxy Repeater serves as both a Proxy Agent and a Proxy Repeater. When a Proxy Repeater sees an unacknowledged broadcast as the message to be delivered by the Proxy Agent, it first sends the message as if it were a Proxy Agent to its Proxy Target, and then relays the message on to the next Proxy Repeater or Proxy Agent.

E.4 Downlink frame format

E.4.1 Overview

This subclause specifies the Repeating parameters (or proxy parameters, shown in Figure 4 and Figure 5. The Downlink proxy parameters contain details describing the entire path from the source to the target device, plus the associated communication parameters (timers, retry count, service type).

There are two forms of Proxy parameters in the Adaptation layer PDU.

When the Proxy Source or a Proxy Repeater is talking directly to the Proxy Target, it uses the form as per Figure E.5. In the Proxy parameters of the AdpPDU, the ProxyAddress field could be either a ProxyDetailAddress or a ProxyUniformAddress field, see Table E.3.

ProxyHeader	ProxyAddress{count}	ProxyCntrl	ProxyTrailer	Adp header	Payload

Figure E.5 – Proxy Repeater proxy message

When the Proxy Source or a Proxy Repeater is talking directly to the Proxy Agent, it uses the form in Figure E.6:

ProxyHeader ProxyTrailer Adp header Payload

Figure E.6 – Proxy Agent proxy message format

When the Proxy Agent is talking directly to the Target, it uses the form in Figure E.7.

Figure E.7 – Proxy Target message format

The various fields are specified in the following subclauses.

E.4.2 Proxy parameters of request frame

E.4.2.1 ProxyHeader

All proxy parameters of the Request frame contain the ProxyHeader. The format of the remainder of the frame is determined based on the 'hop counter' value. If the value is 0, the frame is from Proxy Agent; otherwise, the frame is from Proxy Repeater.

The ProxyHeader is a 16 bit field, with the structure shown in Table E.1.

Table E.1 – ProxyHeader format

Bits	Definition
15-8	Proxy message code. Value 0x4D
7	Use destination subnet. If set, the subnet_ID of the next hop is the same as that of the current hop. NOTE bits 7 and bit 4 are mutually exclusive. If both bits are cleared, then each subnet is specified in the proxy address
6	Use Long timer. If set, the repeater uses a transaction timer computed using the node's receive timer as a base, rather than the standard tx timer.
5	All agent. If set, every repeater is also an agent. Used in cases where the Proxy Target is a broadcast address.
4	Use source subnet. If set, the subnet_ID of the next hop is the same as that of the source of the current device. NOTE bits 7 and bit 4 are mutually exclusive
3-0	Hop counter. Specifies number of Proxy (Uniform) Address fields.
See also E.4.2.2.2 and E.4.2.2.3.	

NOTE If there is no repeater, then the conventional ISO/IEC 14908 message format is used and the ProxyHeader does not apply.

E.4.2.2 Proxy Address

E.4.2.2.1 Overview

The Proxy Address field contains the addresses of the Proxy Repeaters and the Proxy Agent along the path. As all the devices, (Proxy Source, Proxy Repeaters and Proxy Agent) belong to the same domain, the address is fully defined by the pair {Subnet_ID, Node_ID}.

The ProxyAddress (or ProxyUniformAddress) fields are inserted by the Proxy Source in repeater path order, starting with the 2nd ProxyRepeater through the Proxy Agent. The source delivers the message to the first repeater or agent. Therefore, the first repeater (or agent) address is not necessary. The 'hop count' is set to the number of addresses specified.

NOTE The hop count in the ProxyHeader is the number of ProxyRepeaters. For example, if the path is the Source – Repeater1 – Repeater2 – Repeater3 – Agent – Target, then the hop count is 3.

As each Proxy Repeater forwards the message, it removes the address of the next hop and decrements the 'hop counter' field by 1. When the last address is removed, the ProxyTxCtrl field is also removed from the forwarded packet. When the messages arrive at the ProxyTarget, only the ProxyHeader and the Proxy Trailer remain, as a conventional ISO/IEC 14908 message.

This field is empty when repeaters are not involved in the data routing.

E.4.2.2.2 ProxyDetailedAddress form

This form of the address ProxyDetailledAddress format as specified in Table E.2 is used when not all devices in the repeater path are on the same subnet (both b7 and b4 are not set in the ProxyHeader). If they are all on the same subnet, then either b7 or b4 may be set.

Table E.2 – ProxyDetailedAddress format

Bits	Definition
15-8	subnet_ID of repeater/agent.
7	This refers to the use of ISOIEC/14908-3 frequencies. If 0, use primary frequency. If 1, use secondary. This allows the frequency to be different at each hop, since impairments may differ from hop to hop.
6-0	node_ID of repeater/agent.

For each repeater involved in the path, the ProxyDetailedAddress has to be specified, starting with the second repeater.

E.4.2.2.3 ProxyUniformAddress form

This form of the address as specified in Table E.3 of the address is used when all devices in the repeater path are on the same subnet (either b7 or b4 are set in the ProxyHeader). If b7 is set, the repeater uses its subnet to forward to the next hop; if b4 is set, the repeater uses the subnet of the source hop.

Table E.3 – ProxyUniformAddress format

Bits	Definition
7	Alternate frequency – see Table E.2 bit 7 –.
6-0	node_ID of repeater/agent.

E.4.2.3 ProxyTx Control

The ProxyTxCtrl field (see Table E.4), provides information about the retries and timeout. This field is not present when repeaters are not involved in the data routing.

Table E.4 – ProxyTx control format

Bits	Definition
7-4	Retries. Number of retries to be used by each Proxy Repeater NOTE Only the ProxyRepeaters are concerned Not the ProxyAgent
3-0	Timer. Retry timer used by each Proxy Repeater

E.4.2.4 ProxyTrailer

E.4.2.4.1 Overview

The ProxyTrailer field contains the parameters required by the agent to deliver the AdpPDU to the target. Depending on the 'mode' value in the ProxySicb field, the ProxyTrailer can take either of the following forms as shown in Figure E.8 and Figure E.9:

ProxySicb	ProxyTargetAddress

Figure E.8 – Normal ProxyTrailer

ProxySicb	ProxyKey	ProxyTargetAddress

Figure E.9 – Alternate Key Proxy Trailer

E.4.2.4.2 Proxy SICB

The Figure E.10 specifies the Proxy SICB field structure, which specifies how to send to the target. The proxy SICB field is of one byte as in Figure E.10:

b7	b6	b5	b4	b3	b2	b1	b0
Address Type			Alternate Path	Service Type		Mode	

Figure E.10 – Proxy SICB field structure

The Address Type field indicates the type of address used with the values as shown in Table E.5:

Table E.5 – Address Type values

Address type	Value
Unused	0
Proxy Subnet node	1
Proxy Unique node_Id	2
Proxy broadcast	3
Unused	4
Proxy Subnet node compact source	5
Proxy Unique node_Id compact	6
Proxy Subnet node compact destination	7
See also E.4.2.4.4.	

When the device is not yet commissioned, and therefore, has not yet been assigned logical address (Domain_ID, Subnet_ID, and Node_IDs), the address is defined by the 48-bit worldwide unique identifier Unique_Node_ID.

When the device is commissioned, the address form (Domain_ID, Subnet_ID, NodeID) is used.

The Alternate Path field is as defined in ProxyDetailedAddress see E.4.2.2.2.

The Service Type field indicates the kind of Transport layer service used: (unackd, unackd-rpt, ackd, request/response). Table E.6 specifies the value of the Service Types.

Table E.6 – Service types

Service type	Value	Description
ACKD	0	acknowledged
UNACKD_RPT	1	unacknowledged repeat
UNACKD	2	unacknowledged
REQUEST	3	request/response

The Mode field indicates the ProxyTrailer format and is one of the values specified in Table E.7.

Table E.7 – Mode types

Mode	Value	Proxy Trailer
Normal	0	Normal
Zero Sync	1	Normal
Alternate Key	2	Alt Key
Reserved	3	
NOTE Zero Sync and Alternate Key are mutually exclusive.		

When Mode value 1 (Zero Sync) is used, the ProxyAgent synchronizes the request to the ProxyTarget with the zero crossing. A packet of this type is discarded if it cannot be sent at the zero crossing.

When Mode value 2 (Alternate key) is used, the ProxyTrailer contains the Proxy Key for communicating with the Target. See E.4.2.4.3.

E.4.2.4.3 Proxy key

The alternate key format as shown in Figure E.11 is used when the Proxy Agent and Proxy Target do not share the same authentication key. In this case, the Proxy Trailer contains the key type and a key delta value to be used to determine the target key value.

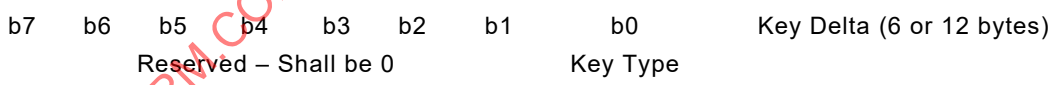

Figure E.11 – Alternate Key format

Where Key Type is one of:

0 – Standard 6 byte key (not used)

1 – Twelve byte key

The bytes in the Key Delta key value are added to the Proxy Agent's key value to determine the target key value.

E.4.2.4.4 Proxy Target Address

As specified in Table E.5, there are 6 types of proxy target addresses. When the device is not yet commissioned, therefore no Domain_ID, Subnet_ID and Node_ID have been assigned as yet, it can be addressed using only types 2 or 6 (Unique_node_Id of 48 bits).

The other types are specified below:

- ProxySubnetNode (Type 1) is made of 2 bytes: the Subnet_ID, a reserved bit, and the Node_ID. See Figure E.12.

Byte0	Byte1.B7	Byte1.B6 – Byte1.B0
Subnet_ID	Reserved	Node_ID

Figure E.12 – ProxySubnetNode Address (Type1)

- ProxyUniqueNodeID (Type 2) is made of 7 bytes specifying the Subnet_ID and the Unique_Node_ID. It is used to address an uncommisioned device. It can be used to address commissioned device as well, but will result in a larger message header. See Figure E.13.

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6
Subnet_ID	Node_ID					

Figure E.13 – ProxyUniqueNode_ID Address (Type 2)

- ProxyBroadcastAddress (Type 3) is made of 2 bytes: the Subnet_ID and the backlog. It is used to address all commissioned devices in the subnet. See Figure E.14.

Byte0	Byte1
Subnet_ID	Backlog

Figure E.14 – ProxyBroadcast Address (Type 3)

- ProxySubnetNodeCompact (Type 5 and Type 7) is made of one byte containing the Node_ID. This is similar to Type 1 but uses the source subnet from the received message. See Figure E.15.

Byte0.B7	Byte0.B6 – Byte0.B0
Reserved	Node_ID

Figure E.15 – ProxySubnetNode Compact Address (Type 5 and 7)

- ProxyUniqueNode_IDCompact (Type 6) (source or destination) is made of 6 bytes defining the Subnet_ID and Unique Node_ID. This is similar to Type 2, but uses subnet 0 (any subnet). See Figure E.16.

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5
Node_ID					

Figure E.16 – ProxyUniqueNodeCompact Address (Type 6)

- Types 0 and 4 are not used.

E.5 Uplink Frame format

E.5.1 Overview

The uplink frame PDU is either the response as returned by the Proxy Target, or the acknowledgement.

E.5.2 Proxy success

When the repeater has successfully forwarded a request or an unackd-rpt message, it returns a proxy successful response consisting of 0x4D value; Table E.8.

Table E.8 – Repeating success response

Byte	Description
0	Proxy success code 0x4D.

E.5.3 Repeating Failure

If any Proxy Repeater is unable to reach the next hop, or the Proxy Agent is unable to reach the Proxy Target due to repeating failure, the following response specified in Table E.9 is returned.

Table E.9 – Repeating Failure

Byte	Description
0	Proxy Failure code 0x4C.
1	Hop count. Count in the chain where the failure occurred.

E.5.4 Authentication failure

If any repeater is unable to reach the next hop, or the agent is unable to reach the target due to authentication issues, the following response specified in Table E.10 is returned:

Table E.10 – Authentication failure code

Byte	Description
0	Authentication Failure code 0x4B.
1	Failure Reason. 0 = Key mismatch

Authentication failure would be indicated by the transport layer similarly to other transaction failures such as timeout.

Annex F
(informative)

ISO/IEC 14908-3 Registration and monitoring of LNAPs

F.1 Scope

This Annex F specifies how the LNAPs are discovered and enrolled inside the neighbourhood PLC network.

According to ISO/IEC 14908-3:2012 PLC neighbourhood network, the NNAP is responsible for creating the neighbourhood network, discovering new LNAPs, determining the optimal path to each LNAP, monitoring for topology changes that require path recalculation, and monitoring for LNAPs that have lost communication. For these purposes, the NNAP uses:

- ATM (Automatic Topology Management) protocol for discovery,
- FCM (Fast Commissioning Message) for enrollment.

F.2 Terms and definitions

F.2.1
commissioning
setting the physical address and security parameters on the device

F.2.2
decommissioning
removing the physical address and security parameters on the device

F.2.3
session
series of transactions each of which consists of a Query ID Request and zero or more Query ID Responses

Note 1 to entry: In the scope of this annex, it is related to discovery session performed using ATM.

F.2.4
test points
LNAPs which are configured for reporting their signal quality in order to indicate failure of a part of the path efficiently about the network performance

F.3 Metering device lifecycle

The management lifecycle of a metering device consists of the following stages:

a) At manufacturing, the LNAP is set with a Unique_Node_ID and the discovery Domain_ID (see F.4.4), and provisioned with the parameters for metrological data measurement and logging as ordered by the customer.

b) The LNAP is physically installed at the utility customer's premises. Initially, it is in the "undiscovered" state. The LNAPs may begin performing metrological functions so that billing data would not be missed before the data collection is initiated by the HES. However, they do not initiate discovery.

c) The NNAP discovers the LNAP using the Automated Topology Management (ATM) protocol. When successful, the LNAP enters the "discovered" state. It is not yet configured with physical address and security parameters, and does not allow metering data to be collected over the PLC network.

d) The NNAP reports the discovered LNAPs to the HES, along with signal quality information and hop count. When multiple NNAPs are connected on the same low-voltage side of a transformer, it is possible for a LNAP to be discovered by multiple NNAPs.

e) The HES assigns the LNAP to the appropriate NNAP according to system-specific criteria, which may take into account signal quality, hop count, capacity of NNAP, and other external criteria like physical location, etc.

f) The NNAP to which the LNAP is assigned to, configures the LNAP with the physical address that comprises the Domain_ID, Subnet_ID and Node_ID and establishes security parameters on the LNAP, because security is always required for commissioning / decommissioning – see Annex D. The LNAP is now in the "commissioned" state. This state allows the NNAP to have DLMS/COSEM data exchange with the LNAP as required by the HES, using secured messages.

g) After that, the NNAP continues to use the ATM protocol to discover new LNAPs and update already discovered LNAPs connectivity status. It continues to monitor the network for topology changes.

h) The NNAP also tests communication with already discovered and commissioned LNAPs in order to validate connectivity, topology changes, and recalculates repeating paths.

i) The NNAP collects data from the commissioned LNAPs and sends commands to them. Each message specifies the optimal repeating path currently determined by the NNAP for the target LNAP.

j) A change in the topology of the electrical network (such as a circuit breaker operation) may cause an extended period of loss of communication between the LNAP and its NNAP. If after that, another NNAP discovers or re-discovers that this LNAP did not communicate with its NNAP for a specified period of time, the LNAP is said to be in an "orphan" state. The latter NNAP reports this status to the HES, which may then reassign the LNAP to the discovering NNAP. Ultimately, the decision to change the NNAP managing the LNAP belongs to the HES, which has to provide all the security parameters before a NNAP will be able to exchange data with the LNAP.

When the HES determines it wants to remove the LNAP from the network, it de-assigns the LNAP from the NNAP. The NNAP then decommissions the LNAP (if it is still able to communicate with it). The LNAP now is again in the "undiscovered" state.

F.4 ATM Protocol

F.4.1 General

The ATM protocol is based on message broadcast to all the nodes that can hear the NNAP via repeaters.

F.4.2 ATM roles

The roles of ATM protocol are:

- automatic discovery of a LNAP at installation;
- automatic re-discovery of a LNAP by another NNAP when the former NNAP can no longer reach the LNAP e.g. due to a topology change;
- automated identification of test points;
- network bandwidth sharing via automatic identification of transmission slots.

F.4.3 Automatic discovery

Automatic discovery allows the NNAP to discover any LNAP device conforming to the protocol (including other NNAPs). The discovery can be made through LNAP repeaters whether or not those LNAP repeaters are commissioned, commissioned in the NNAP's domain or commissioned in some other domain. They only need to be in a global discovery domain – see F.4.4.

The protocol allows the NNAP to send broadcast messages directly or via Proxy Repeaters that ask the Proxy Targets to respond based on certain criteria. Depending on whether the NNAP is trying to discover all devices or only those that are out of communication, it can set the criteria in an appropriate way (see Table F.7).

When attempting to locate nodes that are out of communication, the focus is on finding groups of nodes that were re-located due to topology changes. Thus, it is assumed to be sufficient to attempt to discover a subset of nodes in one domain via a subset of nodes in another domain. Each subset is the set of test points in the domain. The test points are those that are closest to the boundaries of the domain, and thus are best suited for detecting changes at the boundary.

F.4.4 Discovery domain

ATM defines a global discovery domain (DD) using the 6-byte domain ID (0x7A3340F1BCD2). All devices will always be configured in the DD. The DD is configured as an ISO/IEC 14908-3:2012 clone domain so that subnet/node conflicts are not an issue. The key for the DD is the first 6 bytes of the authentication key. All messages are sent unackd.

F.4.5 ATM messages

F.4.5.1 ATM message structure

All ATM messages are based on the same structure consisting of a header and then the message payload.

F.4.5.2 ATM message header

Each ATM ADD (Automatic Device Discovery) message has a header that allows the message to be sent through repeaters, even if those repeaters are not commissioned. This allows a NNAP to discover devices without first performing device commissioning. As such, all addressing is based on unique node_ID and broadcast addressing. Each message contains a repeat chain of unique Node_IDs. The entire repeat chain is passed all the way out and back, so that no transaction tracking is required in the repeaters.

An ATM header contains N addresses, one for each device involved.

The structure of an ATM header is as specified in Table F.1:

Table F.1 – ATM Message header structure

Data	Description	Size (byte)
0x45 or 0x4A	Message or Response code	1
Address position	A one-based number M indicating the position of the destination address in the list of N addresses. See the Note below.	1
Address list	List of N addresses, including the address of the initiator (the NNAP). Each address is of 6 bytes.	6 *N
Path mask	ISO/IEC 14908-1:2012 allows a packet to be transmitted over the primary or an alternate path, where a path refers to the channel frequency. The Path Mask is a bit-map specifying, to each hop's transmitter meaning at each proxy repeater level, which path to use (bit value 0=primary; 1 = alternate).	P bytes
Transaction number	Transaction number. The transaction number is used to match the query ID response to the query ID request.	1
Packet Cycle Width	Cycle width (msec/10) (outbound broadcast only). The Packet Cycle Width is a time interval on the PLC transmission channel and is used by the ISO/IEC 14908 MAC layer.	1
Packet Cycle Count	Number of cycles to randomize over (outbound broadcast only). Range [0, 127].	1
The Path mask has one bit per hop. The first hop is the LSB of the first byte. The number of bytes of the mask is ((N – 2) / 8 + 1), where N is the number of addresses in the address list. Since the address list contains at least the initiator (the NNAP) and the target, N is at least 2. This formula computes the mask size to 1 byte for N = 2 to 9, 2 bytes for N = 10 to 17, etc.		

The header is followed by a message-specific payload as specified in F.4.5.3.

F.4.5.3 ATM message payload

F.4.5.3.1 Overview

Four types of payload are available for ATM management as shown in Table F.2:

Table F.2 – ATM payloads

	Payload identification	Direction
1	ATM-query-ID request	From NNAP to LNAP
2	ATM-query-ID response	From LNAP to NNAP
3	ATM-respond-to-query request	From NNAP to LNAP
4	ATM-respond-to-query response	From LNAP to NNAP

F.4.5.3.2 ATM-query-ID request

The ATM-query-ID message is used to request nodes to respond. Its structure is specified in Table F.3:

Table F.3 – ATM-query-ID request structure

Data	Description	Size (byte)
0x49	Message code.	1
Session ID	Session ID associated with the ATM-respond-to-query message. It is used to match multiple request/response transactions.	1
Hash	Unique node_ID hash. The hash is used to limit the number of responding nodes. NOTE The Hash is a hash range, the upper and lower limits are each a single byte. The receiver computes the hash of its own Unique_Node_ID (adds all bytes in value); then determine whether its hash value is within the range specified. If yes, then the device will respond (assuming it meets all other criteria); if not, then it does not respond. The purpose of this is to reduce the number of devices that can respond to a given discovery request and therefore reduce the chance of collisions.	2
Min Comm Outage	Minimum number of minutes for use with the flag "Out of communication required". See Table F.7. This parameter is used for detecting Orphans.	2
Flags	Flags to limit the responses. See Table F.7.	1

F.4.5.3.3 ATM-query-ID response

The purpose of the ATM-query-ID response is to acknowledge the request. Its structure of the is specified in Table F.4:

Table F.4 – ATM-query-ID response structure

Data	Description	Size (byte)
0x48	Message code	1
PID	Program ID of the device. See B.1.1.	8
Best NNAP UNID	Unique Node ID of the best NNAP. The NNAP UNID returned is that of the NNAP with the best chain quality (fewest hops or best quality in the event of a tie).	6
Quality	Best repeat chain quality value including the NNAP which manages the network. A repeat chain with the smallest number of hops has the best quality. If there are two with the same number of hops to different NNAPs then the NNAP with the best quality is the tie-breaker. Quality has the following format: • 1st byte: hop count, • 2nd byte bit 4...7 (MSB): margin value, • 2nd byte bit 1...3: signal strength divided by 2, • 2nd byte bit 0 (LSB): always 1	2
Flags	See Table F.7.	1

F.4.5.3.4 ATM-respond-to-query request

The ATM-respond-to-query request instructs a device to respond or not to further ATM-query-ID request messages for a given session, based on criteria specified in the Flag; see Table F.7. This message also gives the device a repeat chain quality value to store for this NNAP. It is sufficient for the end device to store only the Unique_Node_ID and Quality value of the NNAP with the best quality. The quality information is based on the number of hops and the signal strength.

Its structure is specified in Table F.5.

Table F.5 – ATM-respond-to-query request structure

Data	Description	Size (byte)
0x47	Message code	1
Session ID	Session ID associated with the ATM-respond-to-query message. It is used to match multiple request/response transactions. The device shall be able to keep up to 8 such NNAP, Session ID pairs. For each session, the device will keep a flag indicating whether to respond to additional queries and it will also store a repeat chain quality value for use in ATM-query-ID responses. It should only be retained for 48 h or until the next reset. Note that because of the limited space for tracking Query ID of the NNAP, it is possible that a device might lose the "respond to query off" context and respond again to subsequent queries. The NNAP shall be capable of detecting this case and backing off by randomizing and trying later.	1
ON/OFF	0: OFF;- send response when queried ON otherwise – do not send response	1
Quality	Chain quality. See Table F.4.	2
Flags	See Table F.7.	1

F.4.5.3.5 ATM-respond-to-query response

The purpose of the ATM-respond-to-query response message is to acknowledge the request. Its structure is specified in Table F.6.

Table F.6 – ATM respond-to-query response structure

Data	Description	Size (byte)
0x46	Message code	1
No other fields follow the message code		

F.4.5.4 Flags

The objective of the flags is to allow selective discoveries, permitting only a given class of LNAPs to take the discovery into account depending on their own state and the values set in the flag.

The Flags are of one byte. This is a bit-mask formed by OR'ing any combination of the values specified in Table F.7:

Table F.7 – Flags

Value	Name	Description
0x01	Ignore query-ID-disable	If this bit is set in a ATM-query-ID request, then the device will ignore any prior ATM-Respond to-query OFF request.
0x02	Out of communication required	The LNAP is to respond only if it is out of communication i.e. it has not received an adaptation layer message from the NNAP in N minutes. In this case the LNAP can receive requests from the NNAP but the NNAP cannot receive response from the LNAP. Therefore, the NNAP should stop sending Adaptation layer messages to the LNAP. NOTE Another NNAP may send Adaptation layer messages to that LNAP so that it can be discovered again.
0x04	Test point required	A device is to respond only if it is a test point (a device on the boundaries of the domain, for detecting topology changes). NOTE Setting test points is performed at the initiative of the NNAP. They are the ones with the highest hop count.
0x08	Commissioned required	A device is to respond only if it is commissioned.
0x10	Uncommissioned required	A device is to respond only if it is decommissioned.
0x40	In communication required	A device is to respond only if it is a LNAP and has received messages from a NNAP (securely or otherwise), or from a meter that has heard from a NNAP, in the last N minutes.
other	reserved	

F.5 Commissioning

F.5.1 Overview

Commissioning is the process of configuring the LNAP with network address and setting the security keys and parameters. This is done using the FCM message, which requires less messaging than the usual ISO/IEC 14908 network management process.

The COSEM-OPEN.request service can only be invoked by the client once the commissioning operation is successfully completed.

F.5.2 Commissioning operations

The commissioning process includes the cryptoContext establishment, and fast commissioning. The primitive that allows performing these operations is specified in D.7.3, D.7.4 and D.7.5 and the messages are specified in D.7.7.

Bibliography

IEC 62056-1-0, *Electricity metering data exchange – The DLMS/COSEM suite – Part 1-0: Smart metering standardization framework*

IEC TS 62056-1-1, *Electricity metering data exchange – Template for DLMS/COSEM communication profile standards*

ANSI C12.19:1997, *IEEE 1377:1997 Utility industry end device data tables*

FIPS PUB 197, *Advanced Encryption Standard (AES)*

NIST Special Publication 800-38B (SP800-38B), *Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication*

NIST Special Publication 800-38C (SP800-38C), *Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality*

NIST the Special Publication 800-108 (SP 800-108), *Recommendation for Key Derivation Using Pseudorandom Functions*

IETF RFC 7539, *ChaCha20 and Poly1305 for IETF Protocols*. Y. Nir, A. Langley May 2015, available at
https://tools.ietf.org/html/rfc7539

IEEE Std 802.11ac-2013, *IEEE Standard for Information technology – Telecommunication and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*

IETF RFC 2119, *Key words for use in RFCs to Indicate Requirement Levels*. S. Bradner, March 1997, available at
https://www.ietf.org/rfc/rfc2119.txt

IETF RFC 3610, *Counter with CBC-MAC (CCM)*. D. Whiting, R. Housley, N. Ferguson September 2003, available at
https://tools.ietf.org/html/rfc3610

IETF RFC 5433, *Extensible Authentication Protocol – Generalized Pre-Shared Key (EAP-GPSK method*. T. Clancy, H. Tschofenig February 2009, available at
https://tools.ietf.org/search/rfc5433

SOMMAIRE

AVANT-PROPOS 107
INTRODUCTION 109
1 Domaine d'application 110
2 Références normatives 110
3 Termes, définitions et abréviations 111
3.1 Termes et définitions 111
3.2 Abréviations 112
4 Environnements de communication ciblés 113
5 Utilisation des couches de communication pour ce profil 114
5.1 Informations relatives à l'utilisation de la Norme spécifiant les couches basses 114
5.2 Structure du profil de communication 114
5.3 Couches inférieures et utilisation 115
5.3.1 Présentation 115
5.3.2 Couche physique 115
5.3.3 Couche liaison de données 115
5.3.4 Couche réseau 115
5.3.5 Couche transport 116
5.4 Mise en correspondance de services et couches adaptation 116
5.4.1 Présentation 116
5.4.2 Couche adaptation 116
5.5 Gestion des enregistrements et des connexions 128
6 Identification et plans d'adressage 128
7 Considérations particulières relatives aux services de couche application 129
7.1 Présentation 129
7.2 Établissement et libération d'associations d'applications: Services ACSE 129
7.2.1 Établissement d'une association d'applications 129
7.2.2 Libération d'une association d'applications 130
7.2.3 Paramètres de service COSEM-OPEN et COSEM-RELEASE 130
7.3 Services xDLMS 130
7.4 Mécanismes de sécurité 130
7.4.1 Sécurité DLMS/COSEM 130
7.4.2 Sécurité de la couche adaptation 130
7.4.3 Sécurité des couches inférieures 131
7.5 Transfert de longs messages d'application 131
7.6 Considérations relatives à l'accès au support, à la bande passante et à la temporisation 131
7.7 Autres considérations 131
8 Configuration et gestion des communications 131
9 Processus d'application COSEM 131
10 Considérations supplémentaires pour l'utilisation de ce profil 132
Annexe A (informative) Exemples 133
Annexe B (normative) Classes d'interface 134
B.1 Classes d'interface 134
B.1.1 ISO/IEC 14908 Identification 134
B.1.2 ISO/IEC 14908 Protocol setup 134

B.1.3 ISO/IEC 14908 Protocol status 135
B.1.4 ISO/IEC 14908 Diagnostic 136
B.2 Relation à l'OBIS 138
Annexe C (informative) Guide de mise en œuvre et chemin de migration 139
C.1 Généralités 139
C.2 Modèle client-serveur 139
C.2.1 Généralités 139
C.2.2 Clients 139
C.2.3 Serveurs 140
C.2.4 Associations d'applications 140
C.3 Objets COSEM 142
C.3.1 Modèle d'objet COSEM 142
C.3.2 Méthodes de référencement 142
C.3.3 Object_list 143
C.3.4 Objets COSEM obligatoires 143
C.3.5 Objets spécifiques à l'application 144
C.3.6 Objets Utility table 144
Annexe D (informative) Suite de sécurité de OSGP-AES-128-PSK 147
D.1 Généralités 147
D.2 Contexte 148
D.2.1 Présentation 148
D.2.2 Hypothèses relatives au système 148
D.2.3 Modèle de menace 148
D.2.4 Objectifs de conception 149
D.2.5 Source 149
D.3 Termes et définitions, abréviations et notation 150
D.3.1 Termes et définitions 150
D.3.2 Abréviations 151
D.3.3 Notation 152
D.3.4 Autres conventions: 152
D.4 Primitives cryptographiques 152
D.4.1 Généralités 152
D.4.2 CMAC 152
D.4.3 CCM 153
D.5 Fonctions cryptographiques 153
D.5.1 Présentation 153
D.5.2 OSGP_KDF: Fonction de dérivation de clés 154
D.5.3 OSGP_MAC: Fonction de code d'authentification de message 154
D.5.4 OSGP_MAC_VERIFY: Fonction de vérification du code d'authentification de message 155
D.5.5 OSGP_AE/OSGP_AD: Fonction de chiffrement/déchiffrement authentifié 155
D.5.6 OSGP_CSPRNG: Générateur de nombres pseudo-aléatoires sécurisé par cryptographie 157
D.6 Clés 158
D.7 Initialisation du canal sécurisé 160
D.7.1 Présentation 160
D.7.2 État du canal sécurisé (CryptoContexte) 160
D.7.3 Flux 160

D.7.4 Négociation de la suite de sécurité 166
D.7.5 Mise en service du LNAP 166
D.7.6 Traitement des erreurs et détection des intrusions 167
D.7.7 Messages 167
D.8 Communication par canal sécurisé 170
D.8.1 Généralités 170
D.8.2 Processus général 171
D.8.3 Communication unicast 173
D.8.4 Communication de diffusion 178
D.9 Téléchargement d'un micrologiciel 180
D.10 Gestion des clés 180
D.10.1 Renouvellement des clés à court-terme, clés uniques du compteur 180
D.10.2 Renouvellement des clés à court terme, clés uniques du domaine 181
D.10.3 UpdateDomainKeysRequest 181
D.10.4 UpdateDomainKeysResponse 182
D.11 Mise à jour des clés à long terme, clés uniques du compteur 183
D.11.1 Généralités 183
D.11.2 Périodes de validité des clés 183
D.12 Messages d'erreur 183
D.12.1 Présentation 183
D.12.2 AuthenticationFailure 183
D.12.3 SequenceError 183
D.13 Considérations de sécurité 184
D.13.1 Justification 184
D.13.2 Recommandations et orientations à l'endroit des implémenteurs 187
Annexe E (normative) Mécanisme de répétition 189
E.1 Vue d'ensemble 189
E.2 Termes et définitions 189
E.3 Spécification du protocole 190
E.3.1 Présentation 190
E.3.2 Adressage 190
E.3.3 Types de service 190
E.3.4 Minuteurs 190
E.3.5 Flux de demande 191
E.3.6 Flux de réponse 191
E.3.7 Authentification 192
E.3.8 Exemples 192
E.3.9 Diffusion 193
E.4 Format de trame de liaison descendante 194
E.4.1 Présentation 194
E.4.2 Paramètres proxy de la trame de demande 194
E.5 Format de trame de liaison montante 200
E.5.1 Présentation 200
E.5.2 Succès du proxy 200
E.5.3 Échec de la répétition 200
E.5.4 Échec d'authentification 200
Annexe F (informative) Enregistrement et surveillance selon l'ISO/IEC 14908-3 des LNAP 201
F.1 Domaine d'application 201

F.2 Termes et définitions 201
F.3 Cycle de vie d'un dispositif de comptage 201
F.4 Protocole ATM 202
F.4.1 Généralités 202
F.4.2 Rôles de l'ATM 202
F.4.3 Découverte automatique 203
F.4.4 Domaine de découverte 203
F.4.5 Messages ATM 203
F.5 Mise en service 207
F.5.1 Présentation 207
F.5.2 Opérations de mise en service 207
Bibliographie 208

Figure 1 – Architecture fonctionnelle de référence 113
Figure 2 – Structure du profil de communication 114
Figure 3 – Structure de la couche adaptation 117
Figure 4 – Structure de la PDU de couche adaptation non protégée 118
Figure 5 – Structure de la PDU de couche adaptation protégée 119
Figure 6 – Services de couche adaptation 120
Figure 7 – Services ADP_DATA client et serveur 124
Figure D.1 – Diagramme de la hiérarchie des clés 158
Figure D.2 – Établissement d'un canal sécurisé 161
Figure D.3 – Demande de défi 162
Figure D.4 – Réponse à la demande de défi 162
Figure D.5 – Demande de mise en service 163
Figure D.6 – Réponse de mise en service 164
Figure D.7 – Établissement d'un CryptoContexte 171
Figure D.8 – Mécanisme de chiffrement authentifié 172
Figure D.9 – Déchiffrement authentifié 173
Figure E.1 – Échange de message sans accusé de réception 192
Figure E.2 –Message requête/réponse 192
Figure E.3 – Erreur lors de la transmission de la requête 193
Figure E.4 – Erreur lors de la transmission de la réponse 193
Figure E.5 – Message proxy du répéteur proxy 194
Figure E.6 – Format de message proxy de l'agent proxy 194
Figure E.7 – Format de message de la cible proxy 194
Figure E.8 –ProxyTrailer normal 196
Figure E.9 – ProxyTrailer de clé complémentaire 197
Figure E.10 – Structure du champ Proxy SICB 197
Figure E.11 – Format de clé complémentaire 198
Figure E.12 – Adresse de ProxySubnetNode (Type 1) 199
Figure E.13 – Adresse de ProxyUniqueNode_ID (Type 2) 199
Figure E.14 – Adresse de ProxyBroadcast (Type 3) 199
Figure E.15 –Adresse de ProxySubnetNodeCompact (Types 5 et 7) 199
Figure E.16 – Adresse de ProxyUniqueNodeCompact (Type 6) 199

Tableau 1 – Champ commande 118
Tableau 2 – SAP client et serveur 128
Tableau 3 – Associations d'applications et échange de données dans le DLMS/COSEM sur profil ISO/IEC14908 129
Tableau C.1 – Paramètres d'association préétablis 141
Tableau C.2 – Plage de base_name pour des catégories de tables 145
Tableau C.3 – base_names pour les tables normalisées BT00 à BT79 145
Tableau D.1– Security logical layers 147
Tableau D.2 – Type de clés et domaine d'application 159
Tableau D.3 – Demande de défi 167
Tableau D.4 – Réponse à la demande de défi 168
Tableau D.5 – Demande de mise en service 169
Tableau D.6 – Structure de la réponse de mise en service 170
Tableau D.7 – Message de demande 174
Tableau D.8 – Structure d'un message de réponse 174
Tableau D.9 – Construction de la valeur unique 174
Tableau D.10 – Construction de données associées 175
Tableau D.11 – Message de diffusion 178
Tableau D.12 – Valeur unique de diffusion 178
Tableau D.13 – Données associées de diffusion 179
Tableau D.14 – Demande de mise à jour des clés de domaine 182
Tableau D.15 – Réponse de mise à jour des clés de domaine 182
Tableau D.16 – Message AuthenticationFailure 183
Tableau D.17 – Message SequenceError 184
Tableau D.18 – Exemples de risque 188
Tableau E.1 – Format de ProxyHeader 195
Tableau E.2 – Format de ProxyDetailedAddress 196
Tableau E.3 – Format de ProxyUniformAddress 196
Tableau E.4 – Format de ProxyTx Control 196
Tableau E.5 – Valeurs de Address Type 197
Tableau E.6 – Types de service 198
Tableau E.7 – Types de mode 198
Tableau E.8 – Réponse succès de la répétition 200
Tableau E.9 – Échec de la répétition 200
Tableau E.10 – Échec d'authentification 200
Tableau F.1 – Structure de l'en-tête de message ATM 204
Tableau F.2 – Charges utiles ATM 204
Tableau F.3 – Structure de la demande ATM-query-ID 205
Tableau F.4 – Structure de la réponse ATM-query-ID 205
Tableau F.5 – Structure de la demande ATM-respond-to-query 206
Tableau F.6 – Structure de la réponse ATM-respond-to-query 206
Tableau F.7 – Indicateurs 207

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

_____

ÉCHANGE DES DONNÉES DE COMPTAGE DE L'ÉLECTRICITÉ – LA SUITE DLMS/COSEM –

Partie 8-8: Profil de communication pour réseaux de la série ISO/IEC 14908

AVANT-PROPOS

1) La Commission Electrotechnique Internationale (IEC) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de l'IEC). L'IEC a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. À cet effet, l'IEC – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de l'IEC"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'IEC, participent également aux travaux. L'IEC collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
2) Les décisions ou accords officiels de l'IEC concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de l'IEC intéressés sont représentés dans chaque comité d'études.
3) Les Publications de l'IEC se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de l'IEC. Tous les efforts raisonnables sont entrepris afin que l'IEC s'assure de l'exactitude du contenu technique de ses publications; l'IEC ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de l'IEC s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de l'IEC dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de l'IEC et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
5) L'IEC elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de l'IEC. L'IEC n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
7) Aucune responsabilité ne doit être imputée à l'IEC, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de l'IEC, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de l'IEC ou de toute autre Publication de l'IEC, ou au crédit qui lui est accordé.
8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de l'IEC peuvent faire l'objet de droits de brevet. L'IEC ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

La Norme internationale IEC 62056-8-8 a été établie par le comité d'études 13 de l'IEC: Comptage et pilotage de l'énergie électrique.

Le texte de cette Norme internationale est issu des documents suivants:

CDV	Rapport de vote
13/1783/CDV	13/1792/RVC

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette Norme internationale.

Ce document a été rédigé selon les Directives ISO/IEC, Partie 2.

Une liste de toutes les parties de la série IEC 62056, publiées sous le titre général *Échange des données de comptage de l'électricité – La suite DLMS/COSEM*, peut être consultée sur le site web de l'IEC.

Le comité a décidé que le contenu de ce document ne sera pas modifié avant la date de stabilité indiquée sur le site web de l'IEC sous "http://webstore.iec.ch" dans les données relatives au document recherché. À cette date, le document sera

- reconduit,
- supprimé,
- remplacé par une édition révisée, ou
- amendé.

IMPORTANT – Le logo *"colour inside"* qui se trouve sur la page de couverture de cette publication indique qu'elle contient des couleurs qui sont considérées comme utiles à une bonne compréhension de son contenu. Les utilisateurs devraient, par conséquent, imprimer cette publication en utilisant une imprimante couleur.

INTRODUCTION

La suite IEC 62056 DLMS/COSEM fournit des normes spécifiques de profil de communication pour les supports de communication destinés au comptage intelligent.

Ces normes de profil de communication spécifient la manière dont le modèle de données COSEM et la couche application DLMS/COSEM peuvent être utilisés sur les couches de protocole inférieures spécifiques au support de communication.

Les normes de profil de communication sont liées aux normes de communication faisant partie intégrante de la suite IEC 62056 DLMS/COSEM ou à une autre norme de communication ouverte.

La présente Norme internationale spécifie le profil de communication DLMS/COSEM utilisant l'ISO/IEC 14908-1:2012 *Information technology – Control network protocol – Part 1: Protocol stack* et l'ISO/IEC 14908-3:2012 *Information technology – Control network protocol – Part 3: Power line channel specification.* Elle s'applique aux équipements installés sur les réseaux de voisinage.

Elle suit les règles définies dans l'IEC 62056-5-3:2017, Annexe A, et dans l'IEC 62056-1-0 ainsi que dans l'IEC TS 62056-1-1 pour sa structure.

ÉCHANGE DES DONNÉES DE COMPTAGE DE L'ÉLECTRICITÉ – LA SUITE DLMS/COSEM –

Partie 8-8: Profil de communication pour réseaux de la série ISO/IEC 14908

1 Domaine d'application

La présente partie de l'IEC 62056 décrit la manière dont la couche application DLMS/COSEM et le modèle d'objet COSEM spécifiés dans l'IEC 62056-5-3:2017, l'IEC 62056-6-1:2017 et l'IEC 62056-6-2:2017 peuvent être utilisés sur les couches inférieures spécifiées dans la série IEC 14908, pour former un profil de communication DLMS/COSEM ISO/IEC14908.

Le présent document fait partie de la série IEC 62056. Sa structure s'aligne sur celle de l'IEC 62056-1-0 et de l'IEC TS 62056-1-1.

L'Annexe A (informative) donne des exemples d'instances représentatives d'échanges de données.

NOTE Cette Annexe A, quoique vide, est incluse et citée en référence par souci de cohérence avec d'autres parties de la série IEC 62056.

L'Annexe B (normative) définit les classes d'interface COSEM et les codes OBIS associés pour la configuration et la gestion du profil de communication DLMS/COSEM pour les réseaux IEC 14908. Ces classes d'interface et ces codes OBIS spécifiés seront intégrés ultérieurement dans l'IEC 62056-6-2 et l'IEC 62056-6-1.

L'Annexe C (informative) fournit un guide de mise en œuvre et spécifie un chemin de migration des applications fondées sur les objets "Utility Tables" (tables de fournisseurs de service d'énergie) vers les applications fondées sur DLMS/COSEM.

L'Annexe D (informative) spécifie la suite de sécurité OSGP-AES-128-PSK à utiliser en option au niveau de la couche adaptation.

L'Annexe E (normative) spécifie le mécanisme de répétition sur le réseau de communications par courants porteurs ISO 14908-3.

L'Annexe F (informative) spécifie l'enregistrement et la surveillance ISO/IEC 14908-3 des LNAP (points d'accès au réseau local).

2 Références normatives

Les documents suivants cités dans le texte constituent, pour tout ou partie de leur contenu, des exigences du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

IEC 62056-5-3:2017, *Échange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 5-3: Couche application DLMS/COSEM*

IEC 62056-6-1:2017, *Échange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 6-1: Système d'identification des objets (OBIS)*

IEC 62056-6-2:2017, *Échange des données de comptage de l'électricité – La suite DLMS/COSEM – Partie 6-2: Classes d'interfaces COSEM*

ISO/IEC 14908-1:2012, *Technologies de l'information – Protocole de réseau de contrôle – Partie 1: Pile de protocole*

ISO/IEC 14908-3:2012, *Technologies de l'information – Protocole de réseau de contrôle – Partie 3: Spécification de canal de courants porteurs*

EN 50065-1, *Transmission de signaux sur les réseaux électriques basse tension dans la bande de fréquences de 3 kHz à 148,5 kHz – Partie 1: Règles générales, bandes de fréquences et perturbations électromagnétiques*

3 Termes, définitions et termes abrégés

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

L'ISO et l'IEC tiennent à jour des bases de données terminologiques destinées à être utilisées en normalisation, consultables aux adresses suivantes:

- IEC Electropedia: disponible à l'adresse http://www.electropedia.org/
- ISO Online browsing platform: disponible à l'adresse http://www.iso.org/obp

NOTE Tous les messages utilisent le format gros boutiste dans le présent document.

3.1 Termes et définitions

3.1.1
domaine
réseau logique constituant une unité pour l'adressage

Note 1 à l'article: Les adresses de sous-réseau (voir ci-après) et de nœud sont attribuées par l'administrateur responsable du domaine et n'ont de sens que dans le contexte du domaine concerné

Note 2 à l'article: Tous les nœuds appartiennent au même domaine pour être en mesure de s'adresser mutuellement.

3.1.2
nœud
abstraction de nœud physique représentant le plus haut niveau de résolvabilité d'adresse sur un réseau

Note 1 à l'article: Un nœud est identifié (adressé) dans un sous-réseau par son identifiant de nœud (logique) appelé Node_ID. Un nœud physique peut appartenir à un ou plusieurs sous-réseaux; dans ce cas, il se voit attribuer un numéro de nœud (logique) pour chaque sous-réseau auquel il appartient. Un nœud physique peut appartenir à deux sous-réseaux au maximum; ces sous-réseaux appartiennent à des domaines différents. Un nœud peut également être identifié (de manière absolue) dans un réseau par son Unique_Node_ID (identifiant de nœud unique) qui est immuable.

3.1.3
sous-réseau
ensemble de nœuds accessibles par le même protocole de couche liaison

Note 1 à l'article: L'identifiant Subnet_ID identifie un sous-réseau dans une adresse logique.

3.1.4
transaction
séquence de messages corrélés

Note 1 à l'article: Par exemple, une requête et la réponse à cette requête font partie d'une transaction unique. Une transaction réussit lorsque tous les messages attendus de tous les nœuds participant à la transaction sont reçus au moins une fois. Une transaction échoue si l'un des messages attendus dans la transaction n'est pas reçu.

3.2 Termes abrégés

Terme abrégé	Signification
AA	Association d'Applications
AARE	Application Association Response (réponse d'association d'applications)
AARQ	Application Association Request (demande d'association d'applications)
ACSE	Application Control Service Element (élément de service de contrôle d'application)
ADD	Automated Device Discovery (découverte automatique de dispositif)
AdpPDU	Adaptation layer PDU (PDU -unité de données de protocole- de couche adaptation)
AL	Application layer (couche application)
AP	Application Process (processus d'application)
APDU	Application Protocol Data Unit (unité de données de protocole d'application)
ASE	Application Service Element (élément de service d'application)
ATM	Automated Topology Management (gestion automatisée de la topologie)
BPSK	Binary Phase Shift Keyed (modulation par déplacement de phase bivalente)
COSEM	Companion Specification for Energy Metering (spécification d'accompagnement pour le comptage de l'énergie)
CRC	Code de redondance cyclique
CSMA	Carrier Sense Medium Access (accès au support à détection de porteuse)
DC	Data Concentrator (concentrateur de données)
DLPDU	Data Link Protocol Data Unit (unité de données de protocole de liaison de données)
DLMS	Device Language Message Specification (spécification de message de langage de dispositif)
FCM	Fast Commission Message (message de mise en service rapide)
KMS	Key Management System (système de gestion de clés)
HES	Head End System (système centralisé de gestion et de télérelevé)
LNAP	Local Network Access Points (points d'accès au réseau local)
MAC	Medium Access Control (contrôle d'accès au support), entité de sous-couche
MAC	Message Authentication Code (code d'authentification de message), en cryptographie
MEP	Multipurpose Expansion Port (accès d'extension multifonctionnel)
NMS	Network Management System (système de gestion de réseau)
NN	Neighbourhood Network (réseau de voisinage)
NNAP	Neighbourhood Network Access Points (points d'accès au réseau de voisinage)
OBIS	OBject Identification System (système d'identification d'objet)
OSGP	Open Smart Grid Protocol (protocole de réseau intelligent ouvert) [a)]
OSI	Open Systems Interconnection (interconnexion de systèmes ouverts)
PID	Program Identifier (identifiant de programme)
PLC	Power Line Carrier (Courant porteur sur ligne d'énergie)
PSK	Pre-Shared Key (clé prépartagée)
SAP	Service Access Point (point d'accès au service)
SDU	Service Data Unit (unité de données de service)
SICB	Service Interface Control Block (bloc de contrôle d'interface de service)
UNID	Unique Node Identifier (Identifiant de nœud unique)
xDLMS	Extended DLMS (DLMS étendu)

a) OSGP est un modèle de travail qui définit une manière dont il convient de mettre en œuvre le comptage intelligent. Il comprend différents éléments et paramètres, notamment la conception et la mise en œuvre de la sécurité, les formats de données, les modèles de données, les fonctionnalités des dispositifs.

4 Environnements de communication ciblés

Le profil de communication DLMS/COSEM pour réseaux série IEC 14908 permet d'échanger des données à distance sur des réseaux de voisinage (NN) entre les points d'accès aux réseaux de voisinage (NNAP) et les points d'accès au réseau local (LNAPs) ou les dispositifs de comptage à l'aide de la technologie PLC à bande étroite BPSK, le réseau de distribution d'électricité basse tension faisant office de support de communication. L'architecture fonctionnelle de référence est représentée à la Figure 1.

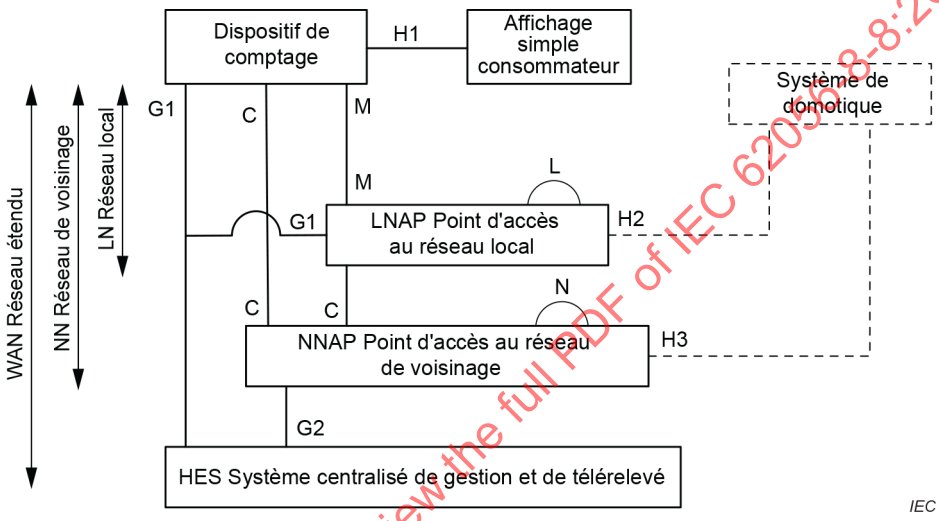

Figure 1 – Architecture fonctionnelle de référence

Les dispositifs de comptage (des compteurs d'électricité, en général) comportent des fonctions d'application et des fonctions de communication. Ils peuvent être reliés directement au NNAP par l'intermédiaire de l'interface C ou à un LNAP par l'intermédiaire d'une interface M, le LNAP étant relié au NNAP par l'intermédiaire de l'interface C. La fonction LNAP peut cohabiter avec les fonctions de comptage.

Un NNAP comprend des fonctions de passerelle et peut comporter des fonctions de concentrateur. Il est relié en amont au système centralisé de gestion et de télérelevé (HES) de comptage à l'aide de supports et protocoles de communication adaptés.

Les dispositifs de comptage et les LNAP peuvent communiquer avec différents NNAP, mais avec un seul NNAP à la fois. Du point de vue de la communication par PLC, le NNAP joue le rôle de passerelle, tandis que les dispositifs de comptage et les LNAP jouent le rôle de points terminaux.

Les NNAP, et de façon similaire les LNAP, peuvent communiquer entre eux, mais cela ne fait pas partie du domaine d'application du présent document, qui ne couvre que l'interface C.

Si le NNAP possède des fonctions de concentrateur, il agit comme un client DLMS/COSEM. Si le NNAP possède uniquement des fonctions de passerelle, le HES joue le rôle d'un client DLMS/COSEM. Les dispositifs de comptage ou les LNAP agissent comme des serveurs DLMS/COSEM. Dans cette spécification, le NNAP est considéré comme jouant le rôle d'un client DLMS/COSEM, ce qui signifie qu'il possède des fonctions de concentrateur. Par conséquent, il est aussi appelé concentrateur de données (DC).

5 Utilisation des couches de communication pour ce profil

5.1 Informations relatives à l'utilisation de la norme spécifiant les couches basses

Le profil de communication DLMS/COSEM pour réseaux de la série ISO/IEC 14908 utilise les services des couches inférieures spécifiées dans l'ISO/IEC 14908-1:2012 et l'ISO/IEC 14908-3:2012.

5.2 Structure du profil de communication

La structure du profil de communication DLMS/COSEM pour les réseaux de la série ISO/IEC 14908 est représentée à la Figure 2.

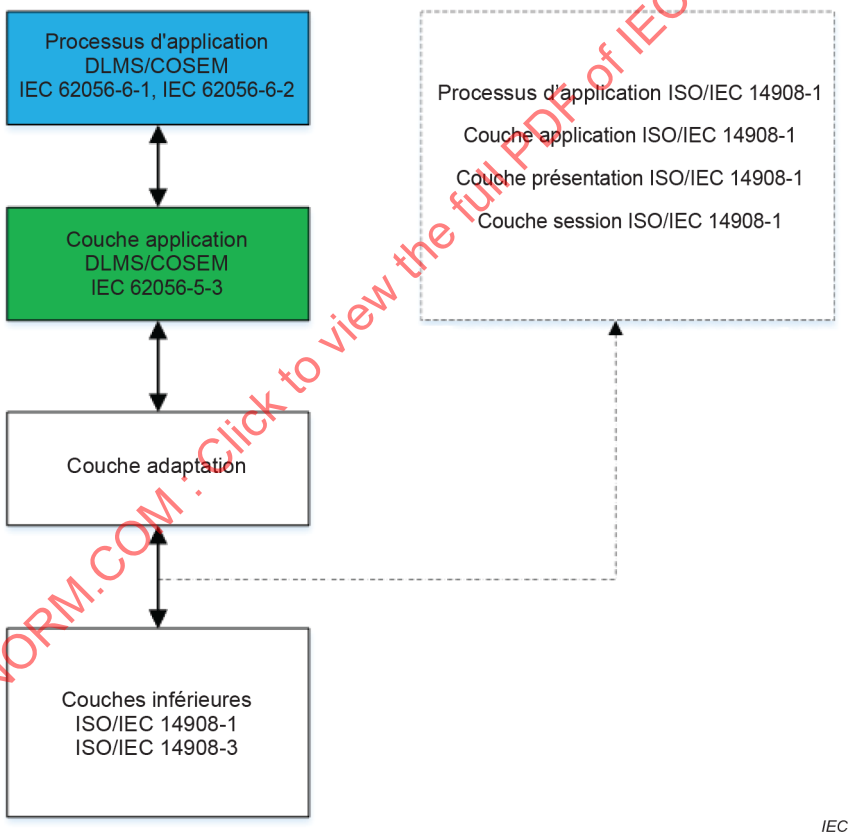

Figure 2 – Structure du profil de communication

Les parties représentées par des traits continus constituent un profil de communication DLMS/COSEM pur pour les réseaux de la série ISO/IEC 14908. Il utilise le modèle de données COSEM spécifié dans l'IEC 62056-6-1:2017 et l'IEC 62056-6-2:2017, la couche application DLMS/COSEM définie dans l'IEC 62056-5-3:2017, ainsi que les couches inférieures spécifiées dans la série de normes ISO/IEC 14908.

Les parties représentées par des traits pointillés ne relèvent pas du domaine d'application du présent document, mais proviennent de la série de normes ISO/IEC 14908.

Tous les messages utilisent le format gros boutiste dans le reste du présent document.

5.3 Couches inférieures et utilisation

5.3.1 Présentation

Les protocoles de couche inférieure sont spécifiés dans l'ISO/IEC 14908-1:2012 et l'ISO/IEC 14908-3:2012. Ils comprennent les couches Physique, Liaison de données, Réseau et Transport selon le modèle OSI.

5.3.2 Couche physique

L'ISO/IEC 14908-1:2012 permet plusieurs types de couches physiques. La couche physique utilisée dans la présente norme est destinée au PLC à bande étroite utilisant la modulation BPSK dans la bande de fréquences A du CENELEC. Cette couche assure l'interface entre le matériel et le support de transmission physique (le réseau de distribution d'électricité). Elle reçoit et transmet des DLPDU entre des nœuds voisins à l'aide de la technologie PLC définie dans l'ISO/IEC 14908-3:2012, en utilisant la modulation BPSK dans la bande de fréquences A du CENELEC.

5.3.3 Couche liaison de données

5.3.3.1 Présentation

La couche liaison de données est composée de deux parties: la sous-couche MAC (Media access control) et la couche Liaison spécifiées aux Articles 6 et 7 de l'ISO/IEC 14908-1:2012.

5.3.3.2 Sous couche MAC

La sous-couche MAC a pour rôle de se raccorder à la couche physique pour assurer tous les services nécessaires à la gestion du support. Elle décharge ainsi la couche liaison des tâches associées au support. Celles-ci incluent l'utilisation de l'algorithme CSMA p-persistant prédictif, la détection de canal en veille, l'estimation du "backlog" (retard) du canal et la gestion des intervalles prioritaires. Voir 6.1 et 6.5 de l'ISO/IEC 14908-1:2012, ainsi que l'Annexe A.2 de l'ISO/IEC 14908-1:2012.

5.3.3.3 Couche liaison

La couche liaison effectue une détection d'erreur au moyen d'un code de redondance cyclique (CRC). Elle n'effectue aucune reprise sur erreur ni reconnaissance d'erreur. Les trames corrompues sont simplement rejetées. Voir l'ISO/IEC 14908-1:2012, Article 7.

5.3.4 Couche réseau

La couche réseau exécute des fonctionnalités récurrentes lorsque la topologie du réseau est une topologie de canal unique (par exemple la topologie du type bus, anneau à jeton ou étoile).

La fonction de routage est utilisée lorsque la topologie du réseau est une topologie arborescente ou topologie simple. Une configuration de réseau doit être d'abord réalisée. Les adresses utilisées sont celles configurées à l'intérieur des routeurs et non celles contenues dans la PDU échangée. Voir l'ISO/IEC 14908-1:2012, Article 8.

Dans le domaine d'application de ce profil de communication, la couche réseau effectue la fonction de reconnaissance d'adresse comme cela est spécifié en 8.6 de l'ISO/IEC 14908-1:2012. Elle transmet à la couche transport les paquets entrants uniquement si le paquet est adressé à l'identifiant de nœud logique ou physique de ce nœud. Le routage est effectué par l'entité de répétition de la couche adaptation (voir l'Annexe E) au lieu de la couche réseau.

Ce profil de communication permet la répétition de la couche application. Le routage de la couche 3 peut être désactivé si celle-ci ne peut l'effectuer aussi efficacement que l'algorithme de répétition.

5.3.5 Couche transport

5.3.5.1 Présentation

La couche transport comprend la sous-couche contrôle des transactions et la couche transport elle-même.

5.3.5.2 Sous couche contrôle des transactions

Le rôle de la sous-couche contrôle des transactions est d'assurer une livraison de données de bout en bout en permettant au destinataire de détecter des paquets en double, en raison de la nature du réseau et du risque de perte de l'accusé de réception. Une transaction comprend une requête et une réponse. Une réponse peut contenir, au choix de l'expéditeur, les données demandées ou simplement un accusé de réception, ou peut être absente. Une réponse n'est pas attendue dans certaines situations (comme une diffusion). Plusieurs mécanismes sont disponibles dans la sous-couche contrôle des transactions pour faciliter les détections: identification de la transaction en cours, limitation du nombre de transactions simultanées, etc. Voir l'ISO/IEC 14908-1:2012, 9.1.

5.3.5.3 Couche transport

La couche transport contrôle le transport de données de bout en bout en vérifiant que la réponse correcte est reçue en fonction de la requête envoyée. En l'absence d'une telle réponse, elle effectue les retransmissions appropriées. Voir l'ISO/IEC 14908-1:2012, Article 10.

5.4 Mise en correspondance de services et couches adaptation

5.4.1 Présentation

La pile de protocole ISO/IEC 14908-1:2012 comporte, en haut de la couche transport, les couches session, présentation et application.

Dans le profil de communication DLMS/COSEM sur les réseaux ISO/IEC 14908, une couche adaptation est située immédiatement au-dessus de la couche transport. La couche adaptation a pour rôle de fournir les éléments nécessaires pour relier correctement la couche application DLMS/COSEM et le modèle d'objet COSEM aux couches inférieures spécifiées dans l'ISO/IEC 14908-1:2012. Elle permet également de choisir la couche application DLMS/COSEM et le processus d'application tels que définis dans l'IEC 62056-5-3:2017 et l'IEC 62056-6-2:2017.

5.4.2 Couche adaptation

5.4.2.1 Généralités

La couche adaptation permet de relier les couches inférieures à la couche application DLMS/COSEM.

Elle prend aussi en charge les processus de découverte, d'enregistrement et de mise en service; voir l'Annexe F.

Les tâches de la couche adaptation dans l'échange de messages de la couche application DLMS/COSEM sont les suivantes:

- fournir et gérer le chemin que doit emprunter le message pour parvenir au destinataire, 5.4.2.6.2.2.2;

- rediriger les messages vers la couche application DLMS/COSEM en fonction de la valeur du champ "Control" (commande) pendant la réception du message, voir 5.4.2.3.1;
- fournir les paramètres d'adressage exigés pour le processus d'application client et le dispositif logique, voir 5.4.2.3.2 points 3 et 4;
- fournir les paramètres permettant de relier correctement le protocole de couche supérieure à la couche transport, voir 5.4.2.5;
- transporter les APDU DLMS/COSEM en utilisant les services ADP_Data;
- spécifier les services de sécurité assurant l'authentification, le chiffrement et la prévention du rejeu. Le principal objectif des services de sécurité de la couche adaptation est de protéger les messages de découverte, d'enregistrement et de mise en service. Se reporter au 7.4.2 pour de plus amples informations sur l'utilisation des services de sécurité de la couche adaptation pour protéger les services ADP-Data.

NOTE Ces services ne dépendent pas du service de sécurité de la couche session de l'ISO/IEC 14908-1:2012 qui assure uniquement l'authentification.

La couche adaptation est fondamentalement un protocole sans connexion. Il est cependant nécessaire d'établir d'abord un CryptoContexte de sécurité (voir l'Article D.7). Ce CryptoContexte est mis en place lors de la mise en service; voir l'Article F.5. Le CryptoContexte peut être renouvelé pendant la durée de vie du dispositif, par exemple à des fins de mise à jour de clés.

5.4.2.2 Structure de la couche adaptation

La Figure 3 représente la structure de la couche adaptation. Bien que la structure de la couche adaptation du NNAP et du LNAP soit identique, le contenu de ses éléments diffère.

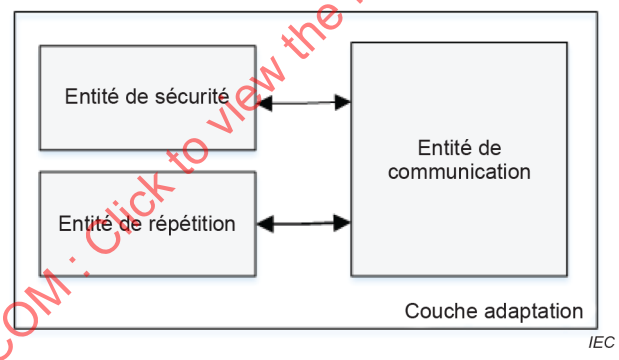

Figure 3 – Structure de la couche adaptation

La couche adaptation contient trois entités:

- une entité de communication en charge – selon la configuration de la couche adaptation – de la construction de la PDU pour les couches inférieures lors de la transmission de données;
- une entité de sécurité qui permet de gérer la protection de l'AdpPDU lorsque la sécurité de la couche adaptation est active;
- une entité de répétition qui permet d'indiquer à l'AdpPDU le chemin que celle-ci doit emprunter pour atteindre de manière optimale le LNAP ciblé.

L'entité de communication est la même du côté NNAP que du côté LNAP. Elle forme, à partir des paramètres de service reçus de la couche application, l'en-tête de couche adaptation lors de la transmission de données. Elle traite, lors de la réception des données, l'en-tête de couche adaptation et présente à la couche application avec la primitive d'indication les données reçues de l'autre nœud.

Un paramètre de configuration indique du côté NNAP si la sécurité de la couche adaptation est active ou non. Lorsque la sécurité de la couche adaptation est active, toutes les clés de chiffrement symétriques associées à l'ensemble des LNAP configurés dans le réseau, et leur identifiant de nœud sont sauvegardées par un système de gestion de clés (KMS). Il appartient aussi à l'entité de sécurité d'effectuer un chiffrement authentifié des AdpPDU sortantes ("chiffrement") et un déchiffrement authentifié, ainsi qu'une prévention du rejeu des AdpPDU entrantes ("déchiffrement"). Voir l'Annexe D pour de plus amples informations.

L'entité de sécurité exécute du côté du LNAP les mêmes fonctionnalités que du côté du NNAP. Le KMS se limite à la gestion de la clé de chiffrement symétrique de la couche adaptation du dispositif associé. La valeur du champ commande reçue du NNAP indique si la sécurité de la couche adaptation est active ou non.

L'entité de répétition indique au message provenant du NNAP le meilleur chemin à emprunter pour atteindre le LNAP ciblé. Les paramètres proxy ne sont disponibles que dans les trames envoyées par le NNAP. Se reporter à l'Annexe E pour de plus amples informations.

5.4.2.3 Structure de la PDU de couche adaptation

5.4.2.3.1 Présentation

La couche adaptation a deux types d'AdpPDU: une AdpPDU non protégée et une AdpPDU protégée, toutes deux identifiées par le champ commande spécifié dans le Tableau 1.

Le Tableau 1 présente les valeurs possibles du champ commande et leur signification dans le domaine d'application du présent document.

Tableau 1 – Champ commande

Valeur	Description
0x20	AdpPDU non protégée
0x21	AdpPDU protégée

Le champ commande fonctionne comme un sélecteur de protocole:

- lorsqu'il est défini sur la valeur 0x20, il indique que la PDU est une PDU de couche application DLMS/COSEM (APDU), la sécurité de couche adaptation étant inactive;
- lorsqu'il est défini sur la valeur 0x21, il indique que la PDU est une APDU DLMS/COSEM, la sécurité de couche adaptation étant active;
- lorsque la valeur est différente de 0x20 ou 0x21, l'APDU n'est donc pas une APDU DLMS/COSEM. Se reporter dans un tel cas au 12.4 de l'ISO/IEC 14908-1:2012. Tout le reste de cet article se rapporte au profil de communication DLMS/COSEM sur profil ISO/IEC 14908.

5.4.2.3.2 Structure de la PDU de couche adaptation non protégée du NNAP

La structure AdpPDU non protégée du NNAP est représentée à la Figure 4. L'AdpPDU contient l'en-tête et la charge utile, ainsi que les paramètres proxy suivants qui sont ajoutés aux paramètres de liaison descendante:

<table>
<tr><td>octet 0</td><td colspan="8"></td><td>octet n</td></tr>
<tr><td colspan="4">Paramètres proxy</td><td colspan="4">En-tête de couche adaptation</td><td>Charge utile</td></tr>
<tr><td>Proxy Header</td><td>Proxy Address(es)</td><td>Proxy Cntrl</td><td>ProxyTx Trailer</td><td>Ctrl Field</td><td>Resv</td><td>DST SAP</td><td>SRC SAP</td><td>DLMS APDU</td></tr>
</table>

Figure 4 – Structure de la PDU de couche adaptation non protégée

Les paramètres proxy contiennent les informations d'adressage exigées pour la transmission du message par le biais d'une série de 1 à 8 répéteurs. Voir l'Article E.4.

L'en-tête de couche adaptation du NNAP est composé des champs suivants:

a) Le champ commande "Ctrl field" a une valeur fixe définie à 0x20;

b) Le champ "Resv" (réservé) est d'un octet et est réservé pour une utilisation ultérieure. Il doit être mis à 0;

c) Le champ "DST SAP" (SAP de destination) contient l'adresse de destination de l'APDU DLMS/COSEM, c'est-à-dire le SAP du dispositif logique lorsque l'APDU DLMS/COSEM est transmise du NNAP au LNAP, et le SAP client lorsque l'APDU DLMS/COSEM est transmise du LNAP au NNAP. La longueur du champ SAP de destination est d'un octet. Voir Tableau 2;

d) Le champ "SRC SAP" (SAP source) contient l'adresse source de la PDU DLMS/COSEM, c'est-à-dire le SAP client lorsque la PDU DLMS/COSEM est transmise du NNAP au LNAP et le SAP du dispositif logique lorsque la PDU DLMS/COSEM est transmise du LNAP au NNAP. La longueur du champ SAP source est d'un octet. Voir Tableau 2.

La charge utile est l'APDU DLMS/COSEM. Elle est la partie restante de l'AdpPDU qui suit le champ "SRC SAP", jusqu'à la fin du message.

5.4.2.3.3 Structure de la PDU de couche adaptation protégée du NNAP

Lorsque la sécurité de la couche adaptation est active, la structure de l'AdpPDU protégée du NNAP est celle représentée à la Figure 5.

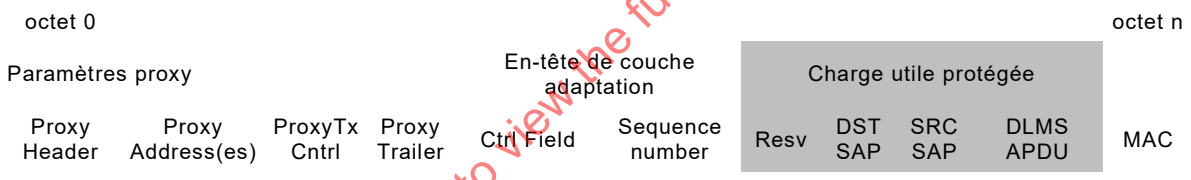

Figure 5 – Structure de la PDU de couche adaptation protégée

où

- le champ commande "Ctrl field" a une valeur fixe définie à 0x21;
- le champ "Sequence number" (numéro de séquence) est de 4 octets. Il s'agit d'une valeur croissante qui référence le message. Elle est remise à 1 chaque fois qu'une nouvelle clé de chiffrement est dérivée. Elle est utilisée pour la détermination de la valeur unique (voir Tableau D.9 et Tableau D.12) et pour la protection contre les attaques par rejeu; voir D.8.4.5;
- la charge utile protégée est le résultat chiffré des champs Resv (réservé), DST SAP (SAP de destination), SRC SAP (SAP source) et de l'APDU DLMS/COSEM (voir Figure 4), concaténés dans cet ordre. Ces champs sont chiffrés et authentifiés à l'aide des algorithmes décrits à l'Article D.4. Pour de plus amples informations, voir l'Annexe D;
- le code d'authentification de message (MAC). La longueur du MAC est de 8 octets. Le MAC est lié à la charge utile protégée Pour de plus amples informations, voir l'Annexe D.

5.4.2.3.4 Structure de la couche adaptation du LNAP

Lorsque la trame est transmise du LNAP au NNAP, la structure de l'AdpPDU est identique à celle spécifiée en 5.4.2.3.2 et en 5.4.2.3.3, sauf que les paramètres proxy sont absents.

5.4.2.4 Longueur maximale de données de la couche adaptation

La longueur maximale de l'AdpPDU est limitée à 228 octets.

Lorsque la protection est inactive, la longueur inclut:

- les paramètres proxy qui peuvent atteindre 48 octets (8 répéteurs au maximum sont autorisés);
- l'en-tête de couche adaptation;
- la charge utile, c'est-à-dire l'APDU DLMS/COSEM.

Lorsque la protection de la couche adaptation est active, la longueur inclut aussi les longueurs du paramètre Sequence number et du MAC.

5.4.2.5 Services de couche adaptation

5.4.2.5.1 Mise en correspondance de services

Le rôle de la couche adaptation est de rediriger l'APDU vers la couche application DLMS/COSEM en fonction de la valeur du champ commande (voir Tableau 1) et de fournir les procédures et services nécessaires pour lier les couches inférieures à la couche application. La Figure 6 présente les services offerts par la couche adaptation.

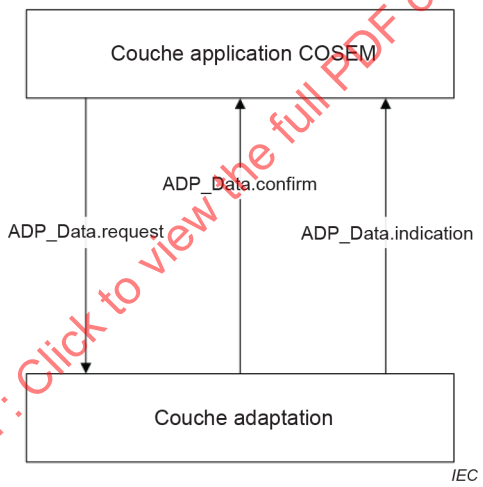

Figure 6 – Services de couche adaptation

5.4.2.5.2 Spécification de service

5.4.2.5.2.1 Généralités

Le protocole de la couche adaptation étant un protocole sans connexion, il ne fournit que des services ADP_Data: ADP_Data.request, ADP_Data.confirm et ADP_Data.indication.

5.4.2.5.2.2 ADP_Data.request

Fonction

Cette primitive est appelée par la couche application DLMS/COSEM pour demander à la couche adaptation d'envoyer une PDU de couche adaptation à l'entité de couche adaptation homologue, selon les procédures de transmission.

Sémantique

La sémantique de cette primitive est la suivante:

```
ADP_Data.request
(
        service_type,
        device_address,
        destination_SAP,
        source_SAP,
        Data
)
```

où:

Le paramètre service_type spécifie le type de TPDU qu'il convient d'utiliser pour le transport des données. Le paramètre service_type est mis en correspondance à partir du champ service_class du paramètre Invoke-Id-And-Priority des services xDLMS ou du paramètre response-allowed de l'APDU InitiateRequest:

- lorsque le champ service_class du paramètre Invoke-Id-And-Priority est de type confirmé ou que le paramètre response-allowed ou l'APDU InitiateRequest est TRUE (vrai), il convient alors que le paramètre service_type soit de type confirm (à confirmation);
- lorsque le champ service_class du paramètre Invoke-Id-And-Priority est de type non confirmé ou que le paramètre response-allowed de l'APDU InitiateRequest est FALSE (faux), il convient alors que le paramètre service_type soit de type unconfirm (sans confirmation);

Le Tableau 3 spécifie la mise en correspondance du champ service-class des paramètres Invoke-Id-And-Priority et response-allowed dans le paramètre service_type.

Le paramètre device_address contient l'adresse ISO/IEC 14908 de destination du dispositif auquel les données doivent être transmises. Elle est composée de subnet_ID et de node_ID. Le paramètre domain_ID est un paramètre de configuration renseigné dans le NNAP lors de la mise en service.

Le paramètre destination_SAP spécifie l'adresse du SAP distant impliqué dans la transmission des données.

Le paramètre source_SAP spécifie l'adresse du SAP à partir duquel le service est demandé.

Le paramètre Data (données) spécifie la PDU à transférer à l'aide des services de couche adaptation.

Utilisation

Cette primitive est utilisée par la couche application DLMS/COSEM lorsqu'elle a des données à transmettre à la couche application de destination.

Dès que la couche adaptation reçoit cette primitive, elle construit l'AdpPDU à mettre à la disposition des couches inférieures en insérant le champ "Control" (commande), le champ "reserved" (réservé), le destination_SAP et le source_SAP au début comme cela est indiqué à la Figure 4, puis applique la protection (voir Figure 5) si la protection est active.

Les paramètres proxy sont également insérés si le NNAP est l'expéditeur. Voir l'Annexe E pour la construction des paramètres proxy.

À l'issue de ce processus, la couche adaptation appelle la primitive de transmission de données de la couche inférieure.

5.4.2.5.2.3 ADP_Data.confirm

Fonction

La primitive ADP_Data.confirm émet une confirmation locale, au niveau de la couche adaptation, à l'achèvement de la transmission des données à la couche inférieure. La portée de cette primitive est uniquement locale. Elle indique à la couche application, ayant appelé une primitive ADP_Data.request, le résultat du traitement de la requête correspondante. Ce résultat peut être SUCCESS (succès) ou FAILED (échec).

Sémantique

La sémantique de cette primitive est la suivante:

```
ADP_Data.confirm
(
        device_address,
        destination_SAP,
        source_SAP,
        status
)
```

Le paramètre device_address contient l'adresse ISO/IEC 14908 de destination du dispositif auquel les données sont transmises. La valeur de ce paramètre est identique à celle fournie dans la primitive ADP_Data.request correspondante.

Le paramètre destination_SAP spécifie l'adresse du SAP distant impliqué dans la transmission des données. La valeur de ce paramètre est identique à celle fournie dans la primitive ADP_Data.request correspondante.

Le paramètre source_SAP spécifie l'adresse du SAP à partir duquel le service est demandé. La valeur de ce paramètre est identique à celle fournie dans la primitive ADP_Data.request correspondante.

Le paramètre status (état) est utilisé pour fournir un résultat local (SUCCESS ou FAILED) de la transmission des données associées à la primitive ADP_Data.request précédemment générée.

Utilisation

Cette primitive est générée en local par la couche adaptation pour l'entité de couche application à titre de réponse à une primitive ADP_Data.request, afin d'informer le demandeur du résultat des procédures de transmission des données.

5.4.2.5.2.4 ADP_Data.indication

Fonction

La primitive ADP_Data.indication est transmise de l'entité de couche adaptation à l'entité de couche application pour l'informer de l'arrivée d'une AdpPDU en provenance de la station homologue ou d'une erreur en provenance de la couche transport. Cette primitive définit le transfert de données de l'entité de couche adaptation à l'entité de couche application.

Sémantique

La sémantique de cette primitive est la suivante:

```
ADP_Data.indication
(
        service_type,
        source_address,
        destination_SAP,
        source_SAP,
        status,
        Data
)
```

Le service_type est un paramètre indiquant le type de service, qu'il s'agisse d'un service de type "confirm" ou "unconfirm".

Le paramètre source_address est l'adresse physique ISO/IEC 14908 du dispositif distant duquel émanent les Data (données). La couche transport fournit ce paramètre à la couche adaptation. Voir la primitive Rcv_Packet (address_pair, pduType, PDU, priority) en 8.3 de l'ISO/IEC 14908-1:2012.

Le paramètre destination_SAP spécifie l'adresse du SAP du dispositif logique lorsque l'APDU DLMS/COSEM est transmise du NNAP au LNAP, et l'identifiant du client (client_ID) lorsque l'APDU DLMS/COSEM est transmise du LNAP au NNAP. Le paramètre destination_SAP fait partie de l'en-tête de couche adaptation fourni par le dispositif distant. Voir 5.4.2.3.2.

Le paramètre source_SAP spécifie l'adresse du SAP du processus d'application ayant envoyé l'APDU DLMS/COSEM. Le paramètre source_SAP fait partie de l'en-tête de couche adaptation fourni par le dispositif distant. Voir 5.4.2.3.2.

Le paramètre status informe la couche application de l'arrivée des données. En cas d'échec, le paramètre status fournit un code d'erreur et le paramètre Data est simplement vide. Voir l'Article D.12.

Le paramètre Data contient l'APDU DLMS/COSEM reçue à l'aide du service de couche adaptation. Voir 5.4.2.3.2.

Utilisation

La primitive ADP_Data.indication est transmise de l'entité de couche adaptation à l'entité de couche application pour l'informer de l'arrivée des Data (données) en provenance d'une entité utilisatrice distante de couche adaptation. Les requêtes du NNAP sont notifiées au LNAP par une primitive ADP_Data.indication au niveau de la couche application du LNAP. À leur tour, les réponses du LNAP aux requêtes du NNAP sont notifiées au NNAP par une indication au niveau de la couche application du NNAP.

5.4.2.6 Spécification du protocole

5.4.2.6.1 Présentation

L'échange de données repose sur le paradigme *client-serveur.* Le client envoie une requête au serveur et, si la requête est de type confirmé, le serveur répond à cette requête comme cela est représenté à la Figure 7.

La transmission de données est initiée par la couche application du client DLMS/COSEM quand cette entité a une requête à envoyer. Lorsque la couche application du client DLMS/COSEM appelle l'ADP_Data.request, elle accompagne la requête de l'adresse de destination du dispositif physique, du SAP client et du SAP serveur. La couche adaptation traite cette requête et appelle les services de couche transport pour sa transmission.

Du côté LNAP, la couche adaptation du serveur reçoit les données de sa couche transport et les traite afin de fournir l'ADP_Data.indication appropriée à sa couche application. Cette indication achemine la requête telle qu'envoyée par le client, à laquelle la couche application du serveur doit répondre, en fonction du type de service d'indication. La réponse, s'il en existe une, est fournie par la couche application à sa couche adaptation au moyen d'une primitive ADP_Data.request. La couche adaptation doit traiter cette réponse et la rendre disponible à la couche transport pour transmission.

Du côté NNAP, la couche adaptation du client reçoit les données de sa couche transport et les traite afin de fournir l'ADP_Data.indication appropriée à sa couche application. Cette indication achemine la réponse telle qu'envoyée par le serveur à sa requête.

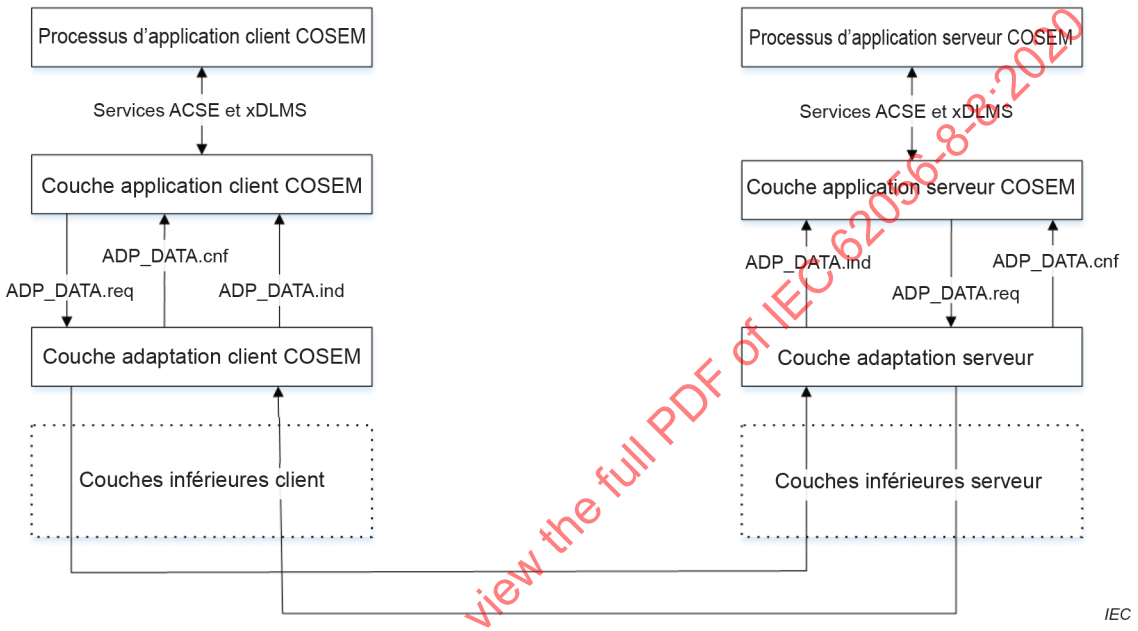

Figure 7 – Services ADP_DATA client et serveur

5.4.2.6.2 Gestion des requêtes

5.4.2.6.2.1 Généralités

Il est nécessaire de répondre à toute requête confirmée reçue au niveau de la couche adaptation (ce qui signifie qu'une ADP_Data.indication doit être appelée par la couche adaptation) avant que la prochaine requête ADP_Data.request ne puisse être appelée par la couche application. La couche transport procède à une annulation de la transmission lorsque sa transaction échoue. Voir 5.3.5.2. Toute nouvelle requête reçue de la couche application, avant réception de la réponse à la précédente, est simplement rejetée.

5.4.2.6.2.2 Envoi de données– côté NNAP

5.4.2.6.2.2.1 En-tête de couche adaptation

Comme indiqué en 5.4.2.2, la couche adaptation du NNAP conserve un paramètre de configuration qui indique si la protection est active ou non. Lorsque la protection est active, elle s'applique à tous les nœuds disponibles dans le sous-réseau. Dans ce cas, un KMS (système de gestion de clés) disponible au niveau du NNAP détient toutes les clés de sécurité associées à chaque nœud du sous-réseau.

Lorsque la couche adaptation du NNAP est invitée à transmettre des données à un LNAP donné, elle insère d'abord le champ commande correspondant, que la protection soit active ou non.

Ensuite, elle insère le champ réservé, le SAP de destination, le SAP source suivi de l'APDU de couche application fournie par la couche application elle-même.

Si la protection n'est pas active, le message créé est transmis à l'entité de répétition pour l'insertion des paramètres proxy. L'AdpPDU construite est mise à la disposition de la couche transport, à l'aide de la primitive appropriée.

Lorsque la protection est active, la couche adaptation construit l'AdpPDU en insérant le champ commande approprié, puis le champ réservé, le SAP de destination, le SAP source suivi de l'APDU fournie par la couche application. Ensuite, la couche adaptation demande la mise en place de la protection du message par l'entité de sécurité. Cette protection comprend le chiffrement puis le calcul du MAC de la partie du message (commençant par le champ réservé suivi des SAP de destination et source, ainsi que de l'APDU DLMS/COSEM fournie par la couche application DLMS/COSEM) à l'aide de la clé symétrique partagée par le NNAP et le LNAP correspondant. À l'issue de cette opération, l'entité de sécurité de la couche adaptation:

- insère le numéro de séquence de la requête à quatre octets immédiatement après le champ commande;
- insère le message chiffré qui est construit; et
- ensuite ajoute le MAC calculé après la charge utile protégée, voir Figure 5.

Le message créé est transmis à l'entité de répétition pour l'insertion des paramètres proxy. L'AdpPDU construite est mise à la disposition de la couche transport, à l'aide de la primitive appropriée.

5.4.2.6.2.2.2 Paramètres proxy

Les échanges reposent sur le principe maître/esclave selon lequel les esclaves (les LNAP) ne sont autorisés à transmettre que sur invitation du NNAP. Un réseau typique est contrôlé par le NNAP, qui joue également le rôle de gestionnaire de la topologie du réseau. Tous les LNAP ont une capacité de répétition. Pour augmenter la portée de communication, l'entité de répétition de la couche adaptation du NNAP peut indiquer aux LNAP situés à l'intérieur du réseau de communication le meilleur chemin qu'il convient que le message d'application emprunte pour atteindre la destination. Ce chemin est spécifié dans les paramètres proxy acheminés par l'AdpPDU. Le message est ensuite transféré d'un saut à l'autre jusqu'à la destination finale. Se reporter à l'Annexe E pour de plus amples informations.

En interne, l'entité de répétition de la couche adaptation du NNAP gère un tableau de répétition qui lui permet de connaître l'ordre des LNAP répéteurs qui sont nécessaires, saut par saut, pour atteindre chaque destination (voir E.2.3) disponible dans le sous-réseau. Pour une adresse donnée du dispositif de destination, l'entité de répétition de la couche adaptation du NNAP extrait les adresses LNAP du tableau de répétition pour les répéteurs, puis construit le champ des paramètres proxy de l'AdpPDU. Les paramètres proxy sont tels que spécifiés par la Figure 4 et la Figure 5. Ceci implique que les champs Proxy Header (en-tête proxy), Proxy Addresses (adresses proxy), Proxy Control (commande proxy) et Proxy Trailer (élément terminal proxy) sont insérés au début de la trame, comme cela est spécifié à l'Article E.3.

5.4.2.6.2.2.3 Appel de service de couche inférieure

Dès lors que les opérations spécifiées en 5.4.2.6.2.2.2 sont terminées, la couche adaptation met le message à la disposition de la couche transport avec le paramètre service_type approprié dans la primitive de service; voir l'ISO/IEC 14908-1:2012, 10.4:

- lorsque le paramètre service_type est de type "confirm", il convient que le type de TPDU requête/réponse soit utilisé;
- lorsque le paramètre service_type est de type "unconfirm", il convient que le type de TPDU Unackd-Rpt soit utilisé;

5.4.2.6.2.2.4 Confirmation par la couche transport

Après avoir mis le message à la disposition de la couche transport pour transmission, la couche adaptation construit et envoie avec succès une primitive ADP_Data.confirm à la couche application chaque fois que l'AdpPDU a été transmise avec succès à la couche transport. Sinon, la couche transport communique un avis d'échec à la couche adaptation.

5.4.2.6.2.3 Envoi de données– côté Serveur

5.4.2.6.2.3.1 En-tête de couche adaptation

Le LNAP répond toujours à une requête confirmée du NNAP après que la couche application du LNAP émet une primitive ADP_Data.request avec un service de type "confirm" (un LNAP ne peut jamais envoyer une requête de type "unconfirm"). La mise en œuvre de la protection est spécifiée par le champ commande:

a) si le champ commande de la requête reçue du NNAP équivaut à une AdpPDU protégée (valeur 0x21, voir Tableau 1), la réponse à envoyer par le LNAP doit être une charge utile protégée;

b) si le champ commande de la requête reçue du NNAP équivaut à une AdpPDU non protégée (valeur 0x20, voir Tableau 1), la réponse à envoyer par le LNAP doit être une charge utile non protégée.

Lorsqu'il est demandé à la couche adaptation du LNAP de transmettre des données au NNAP, elle copie simplement dans sa réponse le champ commande reçu du NNAP, ajoute le champ réservé, puis remplit les champs SAP de destination et SAP source, le tout suivi de la PDU de couche application.

Lorsque la protection n'est pas active, la couche adaptation met le message à la disposition de la couche transport, les paramètres proxy n'étant pas nécessaires pour la transmission des AdpPDU du LNAP au NNAP. Les SAP source et de destination sont fournis dans l'ADP_Data.request émise par la couche application.

Lorsque la sécurité est active, l'entité de communication de la couche adaptation du LNAP exécute la séquence décrite ci-dessus pour la charge utile non protégée, puis transmet les données à l'entité de sécurité qui assure la protection en utilisant les éléments de sécurité disponibles. Chaque serveur stocke, au niveau de la couche adaptation, tous les paramètres de sécurité nécessaires à la protection de la charge utile de la couche adaptation. Il s'agit des paramètres suivants:

- clé de chiffrement de monodiffusion (UCK) et clé de chiffrement de diffusion (BCK) obtenues par dérivation de clés, voir l'Article D.6;
- numéros de séquence pour requête et réception de monodiffusion et numéro de séquence pour réception de diffusion, qui sont de 4 octets chacun. Voir D.8.3.3 et D.8.4.3;
- identifiant unique du compteur et ID de domaine pour le calcul de la valeur unique.

La protection comprend le chiffrement et le calcul du MAC pour la partie du message constituée du champ réservé, des SAP de destination et source et de l'APDU DLMS/COSEM.

L'entité de communication de la couche adaptation concatène pour le champ commande le numéro de séquence de transmission, la charge utile protégée préparée par l'entité de sécurité, puis le MAC. Elle rend ensuite le message disponible pour la couche transport.

5.4.2.6.2.3.2 Paramètres proxy

Les paramètres proxy n'existent pas lorsque le message de réponse est transmis du LNAP au NNAP.

5.4.2.6.2.3.3 Appel de service de couche inférieure

Du côté LNAP, l'entité de communication de la couche adaptation ne traite pas le champ service_type. Toute ADP_Data.request envoyée par la couche adaptation du LNAP est toujours de type "confirm".

5.4.2.6.2.3.4 Confirmation par la couche transport

Après avoir mis les données à la disposition de la couche transport pour transmission, l'entité de communication de la couche adaptation construit et envoie avec succès une ADP_Data.confirm à la couche application chaque fois que l'AdpPDU est transmise avec succès à la couche transport. Sinon, la couche transport communique un avis d'échec à l'entité de communication de la couche adaptation. Voir l'ISO/IEC 14908-1:2012, 11.3.

5.4.2.6.3 Réception de données– côté NNAP

5.4.2.6.3.1 Gestion des paramètres proxy

L'AdpPDU transmise du LNAP au NNAP ne transporte jamais de paramètre proxy.

5.4.2.6.3.2 En-tête de couche adaptation

Lors de la réception, l'entité réceptrice de la couche adaptation commence par l'analyse du champ commande:

a) si le champ commande est égal à l'AdpPDU non protégée (valeur = 0x20), l'entité de communication de la couche adaptation extrait les SAP de destination et source ainsi que l'APDU de l'AdpPDU reçue, puis construit la primitive ADP_Data.indication pour la couche application;

b) si le champ commande est égal à l'AdpPDU protégée (valeur = 0x21), l'entité de communication de la couche adaptation transmet les données à l'entité de sécurité, qui contrôle et supprime la protection. Le contrôle de la protection comprend:

- la vérification du numéro de séquence reçu dans le message, qui doit être supérieur au dernier numéro de séquence stocké dans l'entité de sécurité du NNAP pour le LNAP associé;
- le déchiffrement et la vérification du MAC en utilisant le matériel de sécurité disponible dans les paramètres de configuration du LNAP associé.

Lorsque le contrôle de la protection réussit, l'entité de communication de la couche adaptation se sert des SAP de destination et source et de l'APDU acheminés par la trame pour appeler la primitive ADP_Data.indication afin d'informer la couche application de la disponibilité des données reçues.

Lorsque le contrôle de la protection du message de requête échoue, les données sont simplement ignorées et la couche adaptation crée et envoie une réponse d'erreur à la couche application avec la primitive d'indication. Dans un tel cas, le champ Data (données) est tout simplement vide. Voir Article D.12.

5.4.2.6.4 Réception de données – côté LNAP

5.4.2.6.4.1 Gestion des paramètres proxy

Lorsqu'un répéteur LNAP constate, à chaque saut, que la trame n'est pas à sa destination finale, il utilise sa propre adresse dans la trame et redirige la partie restante de la trame vers le prochain saut, et ainsi de suite, jusqu'à ce que la trame atteigne sa destination finale. Pour de plus amples informations, voir l'Annexe E.

5.4.2.6.4.2 En-tête de couche adaptation

Le protocole de réception est le même au niveau de la couche adaptation de NNAP et de LNAP.

5.4.2.7 Segmentation et réassemblage

La segmentation n'est pas prise en charge. Voir aussi 7.5.

5.5 Gestion des enregistrements et des connexions

La gestion des enregistrements et des connexions est spécifiée dans l'Annexe F.

6 Identification et plans d'adressage

Au cours du processus d'enregistrement, un LNAP obtient du NNAP une adresse du dispositif physique composée de domain_ID (ID de domaine), subnet_ID (ID de sous-réseau) et de node_ID (ID de nœud). Pour l'application de cette spécification, la taille de domain_ID doit être de 3 octets. Voir l'ISO/IEC 14908-1:2012, Articles E.1, E.2 et E.6.

Les adresses source et de destination du dispositif physique sont transportées dans chaque trame à l'aide des services de sous-couche MAC.

En haut de la pile de protocole du côté NNAP, le processus d'application client utilise le SAP client pour son adressage. La même entité physique peut contenir plusieurs processus d'application client. Chacun d'eux est identifié par son SAP.

En haut de la pile de protocole du côté LNAP, le même dispositif physique peut contenir plusieurs dispositifs logiques. Chacun d'eux peut être accessible au niveau de son SAP.

Une association d'applications établie entre un client spécifique et un dispositif logique spécifique est identifiée par le doublet {(Adresse du dispositif physique client, ID client), (Adresse du dispositif physique serveur, ID de dispositif logique serveur)}.

Le Tableau 2 spécifie les valeurs de SAP pour le client et le serveur

Tableau 2 – SAP client et serveur

SAP client	
Aucune station	0x00
Processus de gestion des clients	0x01
Client public	0x10
Ouvert aux affectations de SAP client	0x02… 0x0F et 0x11 à FF
SAP serveur	
Aucune station	0x00
Dispositif logique de gestion	0x01
Réservé pour une utilisation future	0x02… 0x0F
Ouvert aux affectations de SAP serveur	0x10…0x7E
Toutes les stations (Diffusion)	0xFF

7 Considérations particulières relatives aux services de couche application

7.1 Présentation

Le présent article définit les contraintes particulières qui sont liées à l'utilisation de la couche application DLMS/COSEM dans ce profil de communication.

7.2 Établissement et libération d'associations d'applications: Services ACSE

7.2.1 Établissement d'une association d'applications

Pour pouvoir établir une association d'applications (AA) entre le LNAP et le NNAP gérant le sous-réseau, le LNAP doit d'abord être découvert et configuré.

Comme spécifié dans l'IEC 62056-5-3:2017, les AA peuvent être explicitement établies ou préétablies. Pour établir explicitement une AA, le client utilise les services ACSE. Aucune contrainte spécifique ne s'applique.

Les AA à la fois établies et préétablies de manière explicite peuvent être confirmées ou non.

Le Tableau 3 présente les règles permettant d'établir de manière explicite les AA confirmées et non confirmées, le type de services de transfert de données disponibles dans ces AA et les types d'AdpPDU utilisés pour transporter des APDU. Les zones grises de ce tableau représentent des cas qui ne relèvent pas des conditions de fonctionnement normal: ils sont soit non autorisés soit sans objet.

Tableau 3 – Associations d'applications et échange de données dans le DLMS/COSEM sur profil ISO/IEC14908

Établissement d'une association d'applications				Échange de données	
Paramètres de connexion du protocole	**Classe de services COSEM-OPEN**	**Utilisation**	**Type d'associations d'applications établies**	**Classe de service**	**Type de service ADP-Data**
Adresse du dispositif physique: {domain_ID, subnet_ID, node_ID}	Confirmée	AARQ/AARE utilisant des services ADP-Data	Confirmée	Échange de données confirmé autorisé	ADP-Data confirm
				Échange de données non confirmé autorisé	ADP-Data unconfirm
	Non confirmée		Non confirmée	Échange de données confirmé non autorisé	-
				Échange de données non confirmé autorisé	ADP-Data unconfirm

L'échange de données utilisant des services confirmés et/ou non confirmés peut avoir lieu dans une AA confirmée.

L'échange de données utilisant un service confirmé ne peut pas avoir lieu dans une AA non confirmée.

7.2.2 Libération d'une association d'applications

La couche (N-1) de la couche application, c'est-à-dire la couche adaptation dans ce profil, a un protocole sans connexion. L'AA ne peut donc être fermée qu'au moyen des services ACSE A-RELEASE – en appelant la primitive COSEM-RELEASE.request.

7.2.3 Paramètres de service COSEM-OPEN et COSEM-RELEASE

Les paramètres de service du service COSEM-OPEN doivent être utilisés comme décrit ci-après:

Les paramètres Protocol_Connection_Parameters doivent être les suivants:

Identifiant de (protocole) profil	ISO/IEC 14908
Server_subnet_ID	Sous-réseau auquel appartient le dispositif [1)]
Server_node_ID	Identifiant de nœud attribué au dispositif lors de l'enregistrement
Server_SAP	SAP de l'adresse du dispositif logique COSEM
Client_subnet_ID,	Sous-réseau auquel appartient le client [1)]
Client_node_ID	Identifiant de nœud attribué au client
Client_SAP	Identifiant (type) du processus d'application client COSEM

[1)] Voir E.3.2.

Le service du paramètre User_information n'est pas utilisé.

Le paramètre service_class est spécifié dans le Tableau 3.

Le paramètre Use_RLRQ_RLRE de la primitive COSEM-RELEASE.request doit être positionné à TRUE. Le service du paramètre User_information n'est pas utilisé.

7.3 Services xDLMS

Le profil de communication spécifié dans le présent document permet d'utiliser tous les services xDLMS définis dans l'IEC 62056-5-3:2017 dans un contexte d'application où le référencement par nom logique (LN) ou nom abrégé (SN) est utilisé. Mais en raison du principe de transaction des couches inférieures, les services non sollicités ne peuvent pas être utilisés.

7.4 Mécanismes de sécurité

7.4.1 Sécurité DLMS/COSEM

Les mécanismes de sécurité DLMS/COSEM spécifiés dans l'IEC 62056-5-3:2017 et dans l'IEC 62056-6-2:2017 s'appliquent au niveau de la couche application et au niveau du processus d'application. Ils peuvent être utilisés sans restriction.

7.4.2 Sécurité de la couche adaptation

7.4.2.1 Introduction

L'application des fonctions de sécurité de la couche adaptation aux messages ADP_Data est contrôlée par le champ commande de ces messages, voir 5.4.2.3.1.

7.4.2.2 Suite de sécurité

7.4.2.2.1 Présentation

La présent document prend en charge la suite de sécurité OSGP-AES-128-PSK spécifiée en D.2.5.

7.4.2.2.2 OSGP-AES-128-PSK

La suite de sécurité OSGP-AES-128-PSK (voir l'IETF RFC 5433) fournit un canal sécurisé de communication bidirectionnelle, mutuellement authentifiée, entre un NNAP et un LNAP. Le canal sécurisé assure la confidentialité, l'intégrité et l'authenticité des données, ainsi que la protection contre le rejeu.

7.4.3 Sécurité des couches inférieures

Outre la sécurité DLMS/COSEM et la sécurité de la couche adaptation, les couches inférieures peuvent également fournir des fonctions de sécurité. Ces fonctions de sécurité ne relèvent pas du domaine d'application du présent document.

7.5 Transfert de longs messages d'application

Le transfert de longs messages d'application s'effectue au moyen du mécanisme de transfert général de blocs DLMS/COSEM qui prend en charge le transfert confirmé avec diffusion en flux et récupération de blocs, ainsi que le transfert non confirmé avec diffusion en flux. Ce mécanisme peut transférer tout message de grande taille.

Le transfert de blocs spécifiques aux services xDLMS peut aussi être utilisé dans le domaine de la communication point à point.

7.6 Considérations relatives à l'accès au support, à la bande passante et à la temporisation

Aucune considération particulière n'existe au niveau du processus d'application et de la couche application. Toutes les contraintes de temporisation et de bande passante sont gérées au niveau des couches transport et physique.

7.7 Autres considérations

Vide.

8 Configuration et gestion des communications

Les paramètres de configuration des communications sont gérés au moyen des classes d'interface DLMS/COSEM de l'IEC 14908, telles que spécifiées à l'Annexe B. L'Annexe F spécifie la gestion des enregistrements et des connexions.

9 Processus d'application COSEM

Toutes les fonctions, définies dans l'IEC 62056-6-1:2017, l'IEC 62056-6-2:2017 et l'IEC 62056-5-3:2017, utilisant le paradigme demande/réponse peuvent être utilisées sans restriction.

Voir également l'Annexe C qui définit un chemin de migration des applications utilisant des objets "Utility Table" vers des applications fondées entièrement sur COSEM.

10 Considérations supplémentaires pour l'utilisation de ce profil

Vide.

Annexe A
(informative)

Exemples

Vide.

Annexe B
(normative)

Classes d'interface

B.1 Classes d'interface

B.1.1 ISO/IEC 14908 Identification

Les instances de la classe d'interface "ISO/IEC 14908 Identification" permettent d'identifier le dispositif et le réseau auquel le dispositif est connecté.

ISO/IEC 14908 Identification		**0...n**	**class_id = 130, version = 0**			
Attributs		***Type de données***	***Min.***	***Max.***	***Déf.***	**Nom abrégé**
1. logical_name	(statique)	octet-string				x
2. node_Id	(statique)	unsigned	1	127		x + 0x08
3. subnet_ID	(statique)	unsigned	1	255		x + 0x10
4. domain_ID	(statique)	octet-string				x + 0x18
5. program_ID	(statique)	octet-string				x + 0x20
6. unique_node_ID	(statique)	octet-string				x + 0x28
Méthodes spécifiques		***o/f***				

Description de l'attribut	
logical_name	Identifie l'instance d'objet "ISO/IEC 14908 Identification". Voir l'Article B.2.
node_ID	Identification du nœud.
subnet_ID	Identification du sous-réseau auquel le dispositif est connecté.
domain_ID	Identification du réseau. Il doit être de 3 octets.
program_ID	Identifie de manière unique les fonctionnalités et la version du micrologiciel exécuté sur le dispositif. Il doit être de 8 octets.
unique_node_ID	Identification du module de connexion physique connecté au réseau ISO/IEC 14908-1:2012. Il est attribué à chaque module conforme un identifiant (ID) unique de 48 bits dénommé Unique_Node_ID. Cet ID est unique dans le monde et est défini au moment de la fabrication. La valeur de cet identifiant reste inchangée depuis la phase de fabrication.

B.1.2 ISO/IEC 14908 Protocol setup

Les instances de la classe d'interface "ISO/IEC 14908 Protocol setup" permettent de configurer le dispositif ISO/IEC 14908.

ISO/IEC 14908 Protocol setup		0...n	class_id = 131, version = 0			
Attributs		***Type de données***	***Min.***	***Max.***	***Déf.***	**Nom abrégé**
1. logical_name	(statique)	octet-string				x
2. inactivity_timeout	(statique)	long-unsigned				x + 0x08
Méthodes spécifiques		***o/f***				

Description de l'attribut	
logical_name	Identifie l'instance d'objet "ISO/IEC 14908 Protocol setup". Voir l'Article B.2.
inactivity_timeout	Minutes d'absence des messages de la couche adaptation provenant du NNAP avant que le LNAP ne soit considéré comme étant hors communication; voir F.4.3.

B.1.3 ISO/IEC 14908 Protocol status

Les instances de la classe d'interface "ISO/IEC 14908 Protocol status" permettent de connaître l'état du protocole dans le dispositif ISO/IEC 14908.

ISO/IEC 14908 Protocol status		0...n	class_id = 132, version = 0			
Attributs		***Type de données***	***Min.***	***Max.***	***Déf.***	**Nom abrégé**
1. logical_name	(statique)	octet-string				x
2. transmission_errors	(dyn)	long-unsigned				x + 0x08
3. transmit_tx_failure	(dyn)	long-unsigned				x + 0x10
4. transmit_tx_retries	(dyn)	long-unsigned				x + 0x18
5. receive_tx_full	(dyn)	long-unsigned				x + 0x20
6. lost_messages	(dyn)	long-unsigned				x + 0x28
7. missed_messages	(dyn)	long-unsigned				x + 0x30
8. layer2_received	(dyn)	long-unsigned				x + 0x38
9. layer3_received	(dyn)	long-unsigned				x + 0x40
10. messages_received	(dyn)	double-long-unsigned				x + 0x48
11. messages_validated	(dyn)	double-long-unsigned				x + 0x50
Méthodes spécifiques		***o/f***				

Description de l'attribut	
logical_name	Identifie l'instance d'objet "ISO/IEC 14908 Protocol status". Voir l'Article B.2.
transmission_errors	Nombre d'erreurs de transmission qui se sont produites sur le réseau. Une erreur de transmission est détectée au moyen d'une erreur de CRC lors de la réception de paquets ou si la longueur d'un paquet est inférieure à 8 octets. Cela peut être dû à une collision, un support bruyant, un affaiblissement du signal, etc. Voir l'ISO/IEC 14908-1:2012, 13.8.2.
transmit_tx_failure	Nombre d'échecs de réception par le nœud des accusés de réception ou réponses attendus, après le nombre de relances configurées. Voir l'ISO/IEC 14908-1:2012, Article B.8.

transmit_tx_retries	Nombre de relances envoyées par ce nœud. Ne comprend pas les relances utilisées pour les messages envoyés à l'aide du service répété. Voir l'ISO/IEC 14908-1:2012, Article B.8.
receive_tx_full	Nombre de rejets d'un paquet entrant pour cause d'espace insuffisant dans la base de données de transactions. Voir l'ISO/IEC 14908-1:2012, Article B.8.
lost_messages	Nombre de messages adressés au nœud qui ont été rejetés car aucun tampon d'application n'était disponible pour le message. Voir l'ISO/IEC 14908-1:2012, 13.8.2.
missed_messages	Nombre de messages présents sur le réseau mais non reçus car aucun tampon réseau (tampon de paquets) n'était disponible pour les messages ou le tampon réseau était trop petit pour recevoir les messages. Voir l'ISO/IEC 14908-1:2012, 13.8.2.
layer2_received	Nombre de messages de couche 2 reçus par ce nœud. Les messages de couche 2 sont ceux dont le CRC est correct et qui peuvent être adressés à tous les nœuds. Voir l'ISO/IEC 14908-1:2012, Article B.8.
layer3_received	Nombre de messages transmis par la couche 3 du processeur de protocole. Il peut s'agir de mises à jour de variables de réseau, de messages explicites, d'accusés de réception, de relances, de rappels, de messages de service et de tout autre type de message. Voir l'ISO/IEC 14908-1:2012, Article B.8.
messages_received	Nombre de messages reçus, qui sont transmis à la couche adaptation.
messages_validated	Nombre de messages corrects reçus au niveau de la couche adaptation.

B.1.4 ISO/IEC 14908 Diagnostic

Les instances de la classe d'interface "ISO/IEC 14908 Diagnostic" permettent de connaître l'état du dispositif dans le réseau PLC.

ISO/IEC 14908 Diagnostic		**0…n**	**class_id = 133, version = 0**			
Attributs		***Type de données***	***Min.***	***Max.***	***D2f.***	**Nom abrégé**
1. logical_name	(statique)	octet-string				x
2. plc_signal_quality_status	(dyn)	enum				x + 0x08
3. com_module_state	(dyn)	enum				x + 0x10
4. received_message_status	(dyn)	unsigned				x + 0x18
5. no_receive_buffer	(dyn)	long-unsigned				x + 0x20
6. transmit_no_data	(dyn)	long-unsigned				x + 0x28
7. backlog_overflows	(dyn)	long-unsigned				x + 0x30
8. late_ack	(dyn)	long-unsigned				x + 0x38
9. frequency_invalid	(dyn)	long-unsigned				x + 0x40
Méthodes spécifiques		***o/f***				

Description de l'attribut	
logical_name	Identifie l'instance d'objet "ISO/IEC 14908 Diagnostic". Voir l'Article B.2.
plc_signal_quality_status	Fournit l'état de la qualité du signal PLC enum (0) Aucun trafic PLC détecté, (1) Trafic PLC détecté, mais aucun trafic adressé à ce dispositif, (2) Un paquet adressé à ce dispositif a été reçu. La marge de signal est inférieure ou égale à 9 dB, (3) Un paquet adressé à ce dispositif a été reçu. La marge de signal est comprise entre 12 dB et 15 dB, (4) Un paquet adressé à ce dispositif a été reçu. La marge de signal est supérieure ou égale à 18 dB, (5) Le compteur est mis en service mais son inactivity_timeout défini en B.1.2 a expiré. Le dernier paquet adressé à ce compteur avait une marge de signal inférieure ou égale à 9 dB, (6) Le compteur est mis en service mais son inactivity_timeout défini en B.1.2 a expiré. Le dernier paquet adressé à ce compteur avait une marge de signal de 12 dB ou 15 dB, (7) Le compteur est mis en service mais son inactivity_timeout défini en B.1.2 a expiré. Le dernier paquet adressé à ce compteur avait une marge de signal supérieure ou égale à 18 dB. Ce paramètre est mis à jour toutes les fois que le dispositif reçoit un paquet.
com_module_state	Dernier état de réception du module ISO/IEC 14908 de communication sur ligne d'énergie du dispositif: (0, 1) États invalides, (2) Non configuré, (3) Sans application, (4) Configuré, (6) Hors ligne de façon physique, (12) Configuré, hors ligne de façon logicielle, (140) Configuré, en mode dérivation, (255) Pas de réponse lorsque l'état fait l'objet d'une demande. Ce champ est mis à jour à chaque mise sous tension du dispositif, à chaque demande de resynchronisation initiée par le NNAP et à chaque réinitialisation du module de communication. Toute valeur non répertoriée ci-dessus équivaut à un état invalide. Voir l'ISO/IEC 14908-1:2012,13.4 et 13.8.4.

received_message_ status	État du message reçu: b0: message explicite de liaison PLC reçu, b1: réservé, b2: réservé, b3: réservé, b4: réservé, b5 – b7: Phase de l'émetteur-récepteur 0: Absence d'inversion de phase, 1: En phase normale, 2: Plus 120 degrés, phase inversée, 3: Moins 120 degrés, phase normale, 4: Phase de 180 degrés inversée, 5: Plus 120 degrés, phase normale, 6: Moins 120 degrés, phase inversée, 7: Sans objet
no_receive_buffer	Nombre de fois où les couches inférieures ISO/IEC 14908-1:2012 ont détecté un paquet entrant dans le réseau auquel il est adressé, mais le réseau n'est pas équipé de mémoire tampon pour le stocker. Par conséquent, le message n'a pas pu être mis à la disposition de la couche adaptation.
transmit_no_data	Nombre de tentatives de transmission de données aux piles de protocole ISO/IEC 14908-1:2012 par la couche adaptation, mais aucun tampon n'est disponible. Mis à jour selon les occurrences. Voir l'ISO/IEC 14908-1:2012,13.4.
backlog_overflows	Nombre de fois où le compteur "backlog" prédictif de la couche MAC a connu un dépassement. Voir l'ISO/IEC 14908-1:2012, Article B.8.
late_ack	Nombre de fois où un ACK ou une réponse a été reçu(e) pour une transaction déjà terminée. En général, cela se produit en raison de valeurs trop petites du minuteur de transaction ou de l'encombrement du réseau.
frequency_invalid	Nombre de fréquences invalides détectées par les piles de protocole ISO/IEC 14908-1:2012. Mis à jour selon les occurrences.

B.2 Relation à l'OBIS

Pour la configuration et la gestion de l'échange de données à l'aide des réseaux PLC ISO/IEC 14908, au moins une instance de chacune des classes d'interface (IC) suivantes doit être mise en œuvre:

	IC	Identification de l'OBIS					
		A	B	C	D	E	F
ISO/IEC 14908 Identification	130, ISO/IEC 14908 Identification	0	*b*	32	0	0	255
ISO/IEC 14908 Protocol setup	131, ISO/IEC 14908 Protocol setup	0	*b*	32	1	0	255
ISO/IEC 14908 Protocol status	132, ISO/IEC 14908 Protocol status	0	*b*	32	2	0	255
ISO/IEC 14908 Diagnostic	133, ISO/IEC 14908 Diagnostic	0	*b*	32	3	0	255

Annexe C
(informative)

Guide de mise en œuvre et chemin de migration

C.1 Généralités

Cette Annexe informative a pour but de faciliter la mise en œuvre du profil de communication spécifié dans le présent document. À cet effet, elle récapitule quelques principes de base de DLMS/COSEM spécifiés dans l'IEC 62056-5-3:2017 et dans l'IEC 62056-6-2:2017. Elle spécifie un chemin de migration des applications fondées sur des objets "Utility table" de l'ANSI vers des applications fondées sur des objets COSEM, en utilisant les objets COSEM "Utility table".

Elle spécifie également un moyen permettant de réduire le plus possible le nombre d'échanges et de transférer certains longs messages (les object_lists), en tenant compte des limitations imposées par les couches inférieures du profil de communication spécifié dans le présent document. Pour ce faire, des associations d'applications préétablies sont utilisées et la liste des objets COSEM est définie avec leurs base_names (noms de base) qui peuvent être utilisés dans les mises en œuvre réelles.

C.2 Modèle client-serveur

C.2.1 Généralités

Comme spécifié en 4.2 de l'IEC 62056-5-3:2017, DLMS/COSEM repose sur le modèle client-serveur dans lequel les processus d'application (AP) client et serveur communiquent entre eux. Les clients envoient des demandes de service aux serveurs auxquelles les serveurs répondent. Cet échange se déroule dans les associations d'applications, voir C.2.4.

Dans les environnements de communication ciblés par le profil de communication spécifié dans le présent document, les AP client sont situés dans le NNAP et les AP serveur dans les LNAP. Les LNAP peuvent cohabiter avec les dispositifs terminaux. Voir également l'Article 4, Figure 1.

C.2.2 Clients

Plusieurs clients peuvent accéder à un même serveur, chacun utilisant des contextes spécifiques et des droits d'accès spécifiques.

L'IEC 62056-5-3:2017, 4.1.5 spécifie trois types de contextes différents:

- le contexte d'application qui définit le style de référencement des attributs et des méthodes de l'objet COSEM et l'utilisation du chiffrement;
- le contexte d'authentification qui stipule la méthode et les algorithmes permettant d'authentifier le client et le serveur;
- le contexte xDLMS qui définit l'ensemble des services xDLMS et d'autres paramètres à utiliser.

Les droits d'accès déterminent la liste des objets COSEM visibles dans une AA donnée, les privilèges d'accès à leurs attributs et méthodes, ainsi que la protection nécessaire des demandes de service et des réponses.

Au moins un client public doit être mis en œuvre. Son rôle est de révéler la structure interne des serveurs. Il possède un point d'accès au service (SAP) prédéfini; voir Tableau 2.

Voir également l'IEC 62056-5-3:2017, 4.1.14.

C.2.3 Serveurs

Comme spécifié en 4.7 de l'IEC 62056-6-2:2017, le serveur est un dispositif physique qui contient un ou plusieurs dispositifs logiques. Chaque dispositif logique représente un processus d'application COSEM, qui est modélisé par un ensemble particulier d'objets COSEM.

Il doit être mis en œuvre dans chaque dispositif physique, au moins un dispositif logique de gestion pouvant être le seul dispositif logique. Il possède un point d'accès au service (SAP) prédéfini; voir Tableau 2. Voir également l'IEC 62056-5-3:2017, 4.1.9.

C.2.4 Associations d'applications

C.2.4.1 Présentation

Comme spécifié en 4.1.5 de l'IEC 62056-5-3:2017, l'échange d'informations entre les clients et les serveurs se déroule dans les associations d'applications (AA).

Un client peut être capable d'établir des AA avec un ou plusieurs serveurs. De même, un serveur peut être capable de prendre en charge des AA avec un ou plusieurs clients. Ces AA peuvent être utilisées simultanément ou non, en fonction des capacités du client et du serveur.

Comme spécifié en 4.1.5 de l'IEC 62056-5-3:2017, les AA peuvent:

- être explicitement établies ou préétablies;
- être confirmées ou non confirmées;
- utiliser le référencement par nom abrégé (SN- short name) ou par nom logique (LN, logical name) pour les attributs et méthodes de l'objet COSEM avec le chiffrement ou non;
- utiliser plusieurs mécanismes d'authentification;
- utiliser plusieurs contextes xDLMS différents.

Les AA sont modélisées avec les objets "Association", voir l'IEC 62056-6-2:2017, 5.3.3 et 5.3.4. Chaque AA est identifiée par une paire de SAP client et serveur.

C.2.4.2 AA explicitement établies

Les AA explicitement établies sont établies à l'aide du service COSEM-OPEN spécifié en 6.2 de l'IEC 62056-5-3:2017. Ceci implique un échange AARQ/AARE des APDU ACSE pour négocier le contexte d'application, le contexte d'authentification et le contexte xDLMS. Dans le cas d'une authentification de niveau de sécurité élevé, un échange supplémentaire est exigé sur le résultat du traitement du défi par l'homologue.

C.2.4.3 AA préétablies

Comme spécifié en 4.1.4 de l'IEC 62056-5-3:2017, les AA peuvent également être préétablies. Cela signifie que les contextes de l'AA sont prédéfinis et connus du client et du serveur qui souhaitent les utiliser.

L'utilisation des AA préétablies permet d'éviter les échanges préalables à l'établissement des AA. Cependant, plusieurs fonctionnalités ne sont pas disponibles:

- l'authentification des homologues utilisant des mots de passe ou des défis;
- l'identification de l'utilisateur client;
- l'échange de certificats de clé publique lors de l'échange AARQ/AARE.

Voir également l'IEC 62056-5-3:2017, 6.2.

Le Tableau C.1 spécifie une AA préétablie (utilisable, mais non obligatoire) entre le client avec SAP = 0x7D et le dispositif logique de gestion avec SAP = 0x01.

Tableau C.1 – Paramètres d'association préétablis

Paramètre	Valeur
Client SAP (SAP Client)	0x7D
Server SAP (SAP serveur)	0x01
Application context name (Nom de contexte d'application)	Short_Name_Referencing_With_Ciphering joint-iso-ccitt(2) country(16) country-name(756) identified-organisation(5) DLMS-UA(8) application-context(1) context_id(4)}
xDLMS version (Version xDLMS)	0x06
Client Maximum PDU Size (Taille maximale de PDU client)	58
Server Maximum PDU Size (Taille maximale de PDU serveur)	82
Conformance block (Bloc de conformité)	• general-protection (1) • general-block-transfer (2) • read (3) • write (4) • unconfirmed-write (5) • parameterized-access (18)

Les objets visibles dans cette association doivent contenir:

- les objets obligatoires spécifiés en C.3.4;
- les objets "Utility table" spécifiés dans le Tableau C.3;
- les objets relatifs à la sécurité spécifiés en C.3.5 avec leur class_id (ID de classe), version, base_name (nom de base) et leurs droits d'accès;
- tout autre objet exigé pour l'application avec son class_id, version, base_name et ses droits d'accès.

C.2.4.4 AA publique obligatoire

Une AA est obligatoire: il s'agit de l'AA publique entre le client public et le dispositif logique de gestion. Celui-ci doit prendre en charge une AA pour un client public avec un contexte d'application sans chiffrement et avec l'authentification de niveau de sécurité le plus faible (sans sécurité). Son rôle est de prendre en charge la révélation de la structure interne du dispositif physique et la notification d'événements dans le serveur. Voir également l'IEC 62056-6-2:2017, 4.8.5.

NOTE Les notifications par le serveur ne sont pas prises en charge dans le profil de communication spécifié dans le présent document.

Étant donné que l'AA publique ne fournit pas de sécurité, elle ne peut être utilisée que pour accéder à des informations non sensibles.

C.2.4.5 Autres AA

En plus de l'AA publique obligatoire et en fonction des exigences, des AA explicitement établies ou préétablies peuvent être spécifiées à l'aide du contexte d'application, du contexte d'authentification, du contexte xDLMS et de object_list (c'est-à-dire la liste d'objets).

C.3 Objets COSEM

C.3.1 Modèle d'objet COSEM

Le modèle d'objet COSEM est spécifié dans l'IEC 62056-6-2:2017.

Comme spécifié en 4.1 de l'IEC 62056-6-2:2017, un objet COSEM est un ensemble d'attributs et de méthodes. Les attributs représentent les caractéristiques d'un objet. La valeur d'un attribut peut affecter le comportement d'un objet. Le premier attribut d'un objet est le logical_name (c'est-à-dire le nom logique). Il fait partie de l'identification de l'objet. Un objet peut proposer un certain nombre de méthodes pour examiner ou modifier les valeurs des attributs.

Les objets qui partagent des caractéristiques communes sont généralisés en une classe d'interface, identifiée par un class_id. Une classe d'interface peut avoir une ou plusieurs versions.

Les attributs et méthodes des objets COSEM doivent être accessibles pour l'échange des données.

C.3.2 Méthodes de référencement

Comme spécifié en 4.2.4.3 de l'IEC 62056-5-3:2017, deux méthodes (styles) de référencement sont disponibles pour référencer les attributs et méthodes des objets COSEM:

- le référencement par nom logique (LN); et
- le référencement par nom abrégé (SN).

Le client utilise toujours le référencement LN. Le serveur peut utiliser le référencement LN ou le référencement SN ou les deux.

Lorsque le référencement LN est utilisé, les attributs et méthodes de l'objet COSEM sont accessibles par les demandes de services xDLMS à l'aide du référencement LN: il s'agit des services GET, SET, ACTION et ACCESS qui sont spécifiés du 6.6 au 6.9 de l'IEC 62056-5-3:2017.

Les demandes de service LN contiennent un descripteur d'attribut ou de méthode composé de class_id, logical_name (c'est-à-dire du code OBIS) et de l'ID d'attribut ou de méthode. Un descripteur d'attribut ou de méthode est exprimé sur 9 octets.

Lorsque le référencement SN est utilisé, chaque attribut et méthode de l'objet COSEM est mis en correspondance avec une variable dénommée COSEM, exprimée sur deux octets.

Comme spécifié à l'Article 8 de l'IEC 62056-5-3:2017, les noms abrégés sont de type Integer16 avec les trois derniers bits définis à 0. La plage de noms abrégés va de 0x0008 à 0xFFF8 et 8 191 noms de variable sont disponibles.

L'attribut logical_name de chaque instance d'objet COSEM est mis en correspondance avec un base_name spécifique à la mise en œuvre, qui peut être récupéré par la lecture de l'attribut object_list des objets "Association SN" (SN d'association). Le base_name est normalisé pour certains objets. Les base_names peuvent également être définis dans des spécifications d'accompagnement.

Les décalages relatifs aux autres attributs et méthodes sont spécifiés dans les spécifications des classes d'interface COSEM.

Les attributs et méthodes de l'objet COSEM sont ensuite accessibles par les demandes de services xDLMS à l'aide du référencement SN: il s'agit des services xDLMS Read ou Write qui sont spécifiés du 6.12 au 6.14 de l'IEC 62056-5-3:2017. Les demandes de service SN incluent le nom de base.

Comme mentionné ci-dessus, le client utilise toujours le référencement LN. Par conséquent, si le serveur utilise le référencement SN, l'élément de service d'application (ASE) Client SN_Mapper met en correspondance dans le client les primitives de service LN avec la primitive de service SN, et inversement. Les mises en correspondance sont spécifiées du 7.3.9 au 7.3.11 de l'IEC 62056-5-3:2017.

Les demandes de service utilisant le référencement par nom abrégé sont plus courtes que celles utilisant un référencement par nom logique. Cependant, cette différence n'existe que dans les demandes de service, puisque les descripteurs d'attribut ou de méthode ne sont présents que dans la requête.

D'autre part, du côté client, une traduction entre les APDU utilisant les référencements par LN et SN doit être effectuée pour chaque demande et réponse.

Exemple: Une APDU GET-request contenant le COSEM-Attribute-Descriptor (descripteur d'attribut COSEM) doit être traduite en une APDU Read.request contenant le nom de la variable avec laquelle le COSEM-Attribute-Descriptor a été mis en correspondance.

C.3.3 Object_list

Pour pouvoir communiquer avec le serveur, il est nécessaire pour le client de connaître la méthode de référencement des attributs et méthodes des objets COSEM dans les demandes de service.

Il doit également connaître:

- la liste des objets COSEM accessibles dans chaque AA;
- le base_name de chaque objet, dans le cas du référencement SN; et
- les droits d'accès de chaque attribut et méthode.

La lecture des informations peut s'effectuer à partir des objets "Association SN" (SN d'association) ou "Association LN" (LN d'association); voir 5.3.3 et 5.3.4 de l'IEC 62056-6-2:2017.

L'object_list (liste d'objets) peut également être défini dans des spécifications d'accompagnement propres au projet.

C.3.4 Objets COSEM obligatoires

Comme spécifié en 4.8.4 de l'IEC 62056-6-2:2017, les objets suivants doivent être présents dans chaque dispositif logique COSEM. Ils doivent être accessibles pour GET/Read dans toutes les AA avec ce dispositif logique:

- objet Nom de dispositif logique COSEM;
- objet "Association" actuel, "Association SN" ou "Association LN", conformément à la méthode de référencement utilisée.

S'il existe plusieurs dispositifs logiques, un objet "SAP Assignment" (Affectation de SAP) doit être présent dans le dispositif logique de gestion. Dans ce cas, il n'est pas nécessaire que l'objet Nom de dispositif logique COSEM soit présent.

Les base_names réservés des objets obligatoires sont spécifiés en 4.3 de l'IEC 62056-6-2:2017.

C.3.5 Objets spécifiques à l'application

Les autres objets COSEM peuvent être mis en œuvre selon la fonctionnalité exigée.

Pour prendre en charge la protection cryptographique des APDU xDLMS, les objets suivants sont nécessaires:

- Les objets "Security setup" (établissement de la sécurité), voir l'IEC 62056-6-2:2017, 5.3.7;
- Les objets "Invocation counter" (compteur d'appels), voir l'IEC 62056-6-2:2017, 6.2.33;

Pour prendre en charge la protection des données COSEM, les objets "Data protection" (protection des données) sont utilisés (voir l'IEC 62056-6-2:2017, 5.3.9).

C.3.6 Objets Utility table

Le 5.2.7 de l'IEC 62056-6-2:2017 spécifie les objets "Utility table" (table de fournisseurs de services d'énergie) permettant d'encapsuler les tables de fournisseurs de services d'énergie de l'ANSI C12.19 Utility Industry End Device Data Tables AKA, connu aussi sous le nom d'Utility Tables. Chaque table de fournisseurs de services d'énergie est représentée par une instance de cette IC, identifiée par son nom logique. Les codes OBIS des objets "Utility table" sont spécifiés en 6.2.36 de l'IEC 62056-6-2:2017.

L'utilisation des objets "Utility table" fournit un chemin de migration des applications fondées sur les tables de l'ANSI C12.19 vers des applications fondées sur DLMS/COSEM: les objets "Utility table" peuvent coexister avec d'autres objets COSEM dans la même AA ou dans des AA différentes. Les données contenues dans les tables peuvent être accessibles par des services DLMS/COSEM.

L'ANSI C12.19 spécifie:

- 2 048 tables normalisées (plage d'identifiants de table: 0 à 2 047);
- 2 048 tables de fabricants (plage d'identifiants de table: 2 048 à 4 095);
- 2 048 tables normalisées en cours (plage d'identifiants de table: 4 096 à 6 143);
- 2 048 tables de fabricants en cours (plage d'identifiants de table: 6 144 à 8 191).

Il existe également des tables définies par l'utilisateur.

Pour utiliser les tables avec le référencement SN, les attributs des objets "Utility table" doivent être mis en correspondance avec des noms abrégés. Chaque objet "Utility table" ayant 4 attributs, la mise en correspondance de tous les attributs des tables normalisées et de fabricant avec les noms abrégés nécessiterait 32 640 variables nommées, mais seules 8 191 sont disponibles. Cela limite le nombre de tables ou d'objets "Utility table" accessibles à l'aide du référencement SN.

Les base_names des tables de fournisseurs de services d'énergie mis en œuvre peuvent être récupérés par la lecture de l'attribut object_list de l'objet "Association SN" qui modélise l'AA avec le dispositif logique mettant en œuvre ces tables.

Le présent document spécifie des base_names pour des objets "Utility table" choisis, afin d'éviter la lecture de l'attribut object_list.

Le Tableau C.2 spécifie les plages de base_name (nom de base) disponibles pour les quatre catégories de tables de fournisseurs de services d'énergie.

Tableau C.2 – Plage de base_name pour des catégories de tables

Catégorie de table	Nom de base de début	Nom de base de fin	ID de table	Objets COSEM	
				De	À
Tables normalisées	0x9000	0x9760	0000 – 79	0-0:65.0.0.255	0-0:65.0.79.255
Tables de fabricants	0x9800	0xAFF8	0000 – 2047	0-0:65.16.0.255	0-0:65.31.127.255
Tables normalisées en cours	0xB000	0xB7E0	0000 – 2047	0-0:65.32.0.255	0-0:65.47.127.255
Tables de fabricants en cours	0xB800	0xCFF8	0000 – 2047	0-0:65.48.0.255	0-0:65.63.127.255

Le Tableau C.3 définit le nom de base pour les premières tables normalisées, nommées BT00 à BT79.

Dans le domaine d'application du présent document, la lecture de object_list depuis le serveur peut être évitée si cet objet est connu d'avance, c'est-à-dire:

- toutes ces tables normalisées et uniquement ces tables sont utilisées;
- le base_name de toutes les autres tables est défini dans la spécification d'accompagnement;

  NOTE Par hypothèse, chaque attribut des objets "Utility table" est mis en œuvre et les droits d'accès sont définis dans la spécification d'accompagnement.

- le base_name, la version de interface_class et les droits d'accès à chaque attribut et méthode de tous les autres objets COSEM sont définis dans la spécification d'accompagnement.

Tableau C.3 – base_names pour les tables normalisées BT00 à BT79

ID de table	Code OBIS	base_name	Nom
BT00	0-0:65.0.0.255	0x9000	General Configuration (Configuration générale)
BT01	0-0:65.0.1.255	0x9020	General Manufacturer Identification (Identification générale du fabricant)
BT02	0-0:65.0.2.255	0x9040	Device Nameplate (Plaque signalétique du dispositif)
BT03	0-0:65.0.3.255	0x9060	End Device Mode Status (État du mode du dispositif terminal)
BT04	0-0:65.0.4.255	0x9080	Pending Status (État d'attente)
BT05	0-0:65.0.5.255	0x90A0	Device Identification (Identification du dispositif)
BT06	0-0:65.0.6.255	0x90C0	Utility Information (Informations relatives aux fournisseurs de service d'énergie)
BT07	0-0:65.0.7.255	0x90E0	Procedure Initiate (Initiation de la procédure)
BT08	0-0:65.0.8.255	0x9100	Procedure response (Réponse de la procédure)
BT10	0-0:65.0.10.255	0x9120	Dimension Source Limiting (Limitation de la source de dimensionnement)
BT11	0-0:65.0.11.255	0x9140	Actual sources limiting (Limitation des sources réelles)
BT12	0-0:65.0.12.255	0x9160	Unit of Measure Entry (Unité d'entrée de mesure)
BT13	0-0:65.0.13.255	0x9180	Demand Control (Contrôle de la demande)
BT15	0-0:65.0.15.255	0x91A0	Constants (Constantes)
BT16	0-0:65.0.16.255	0x91E0	Source definition (Définition source)
BT17	0-0:65.0.17.255	0x9200	Alternate source definition (Définition source alternative)
BT20	0-0:65.0.20.255	0x9220	Dimension Register (Registre des dimensions)
BT21	0-0:65.0.21.255	0x9240	Actual Register (Registre réel)
BT22	0-0:65.0.22.255	0x9260	Data Selection (Sélection de données)
BT23	0-0:65.0.23.255	0x9280	Current Register Data (Données du registre courant)
BT24	0-0:65.0.24.255	0x92A0	Previous Season Data (Données de la session précédente)

ID de table	Code OBIS	base_name	Nom
BT25	0-0:65.0.25.255	0x92C0	Previous Demand Reset (Réinitialisation de la demande précédente)
BT26	0-0:65.0.26.255	0x92E0	Self-Read Data (Données d'autolecture)
BT27	0-0:65.0.27.255	0x9300	Present Register Selection (Sélection du registre actuel)
BT28	0-0:65.0.28.255	0x9320	Present Register Data (Données du registre actuel)
BT30	0-0:65.0.30.255	0x9340	Dimension Display (Affichage des dimensions)
BT31	0-0:65.0.31.255	0x9360	Actual Display (Affichage réel)
BT32	0-0:65.0.32.255	0X9380	Display Source (Source d'affichage)
BT33	0-0:65.0.33.255	0x93A0	Primary Display List (Liste d'affichage principale)
BT40	0-0:65.0.40.255	0x93C0	Dimension Security Limiting (Limitation de la sécurité de dimensionnement)
BT41	0-0:65.0.41.255	0x93E0	Actual Security Limiting (Limitation réelle de la sécurité)
BT42	0-0:65.0.42.255	0x9400	Security (Sécurité)
BT43	0-0:65.0.43.255	0x9420	Default Access Control (Contrôle d'accès par défaut)
BT44	0-0:65.0.44.255	0x9440	Access Control (Contrôle d'accès)
BT45	0-0:65.0.45.255	0x9460	Key (Clé)
BT50	0-0:65.0.50.255	0x9480	Dimension Time and TOU (Dimension, Temps et TOU)
BT52	0-0:65.0.52.255	0x94A0	Clock (Horloge)
BT53	0-0:65.0.53.255	0x94C0	Time Offset (Décalage horaire)
BT54	0-0:65.0.54.255	0x94E0	Calendar (Calendrier)
BT55	0-0:65.0.55.255	0x9500	Clock State (État d'horloge)
BT56	0-0:65.0.56.255	0x9520	Time remaining table (Table du temps restant)
BT60	0-0:65.0.60.255	0x9540	Dimension Load Profile (Profil de charge de dimensionnement)
BT61	0-0:65.0.61.255	0x9560	Actual Load Profile (Profil de charge courant)
BT62	0-0:65.0.62.255	0x9580	Load Profile Control (Contrôle du profil de charge)
BT63	0-0:65.0.63.255	0x95A0	Load Profile Status (État du profil de charge)
BT64	0-0:65.0.64.255	0x95C0	Load Profile Data 1 (Données de profil de charge 1)
BT65	0-0:65.0.65.255	0x95E0	Load Profile Data 2 (Données de profil de charge 2)
BT66	0-0:65.0.66.255	0x9600	Load Profile Data 3 (Données de profil de charge 3)
BT67	0-0:65.0.67.255	0x9620	Load Profile Data 4 (Données de profil de charge 4)
BT70	0-0:65.0.70.255	0x9640	Dimension Log (Journal de dimensionnement)
BT71	0-0:65.0.71.255	0x9660	Actual Log (Journal réel)
BT72	0-0:65.0.72.255	0x9680	Event Identification (Identification des événements)
BT73	0-0:65.0.73.255	0x96A0	History Log Control (Historique de contrôle du journal
BT74	0-0:65.0.74.255	0x96C0	History Log Data (Historique des données du journal)
BT75	0-0:65.0.75.255	0x96E0	Event Log Control (Contrôle du journal des événements)
BT76	0-0:65.0.76.255	0x9700	Event Log Data (Données du journal des événements)
BT77	0-0:65.0.77.255	0x9720	Event Log and Signatures Enable (Activation des signatures et du journal des événements)
BT78	0-0:65.0.78.255	0x9740	End Device Program State Table (Tableau d'état des programmes du dispositif terminal)
BT79	0-0:65.0.79.255	0x9760	Event Counters Table (Tableau des compteurs d'événements)

Annexe D
(informative)

Suite de sécurité de OSGP-AES-128-PSK

D.1 Généralités

La suite de sécurité OSGP-AES-128-PSK fournit un canal sécurisé de communication bidirectionnelle entre un NNAP (DC) et un LNAP (compteur). Le canal sécurisé assure la confidentialité des données, l'intégrité des données et de leur l'origine, ainsi que la protection contre le rejeu. Il peut être utilisé pour la communication de diffusion et la communication unicast (ou monodiffusion).

La suite de sécurité OSGP-AES-128-PSK est spécialement conçue pour les réseaux soumis à des contraintes, composés de dispositifs intégrés bas de gamme hérités.

Elle est constituée de trois couches logiques (1 à 3) présentées au Tableau D.1.

Tableau D.1– Security logical layers

Couche logique	Nom	Contenu
4	ADP	Protocole de couche supérieure
3	Protocole de canal sécurisé	Initialisation du canal sécurisé Communication unicast par canal sécurisé Communication de diffusion par canal sécurisé
2	Fonctions cryptographiques	OSGP_KDF OSGP_MAC OSGP_MAC_VERIFY OSGP_AE OSGP_AD
1	Primitives cryptographiques	AES-128 CCM CMAC

Chaque couche logique utilise le service de la couche logique située en dessous (le cas échéant) et fournit du service à la couche logique située au-dessus (le cas échéant). Par exemple, la couche logique du protocole de canal sécurisé utilise la fonction OSGP_AE fournie par la couche logique des fonctions cryptographiques pour protéger les messages. Cette couche logique des fonctions cryptographiques est elle-même utilisée pour passer une communication par canal sécurisé à la couche logique adaptation située au-dessus.

La suite OSGP-AES-128-PSK exige l'établissement sécurisé d'une clé initiale prépartagée. La méthode d'établissement de cette clé ne relève pas du domaine d'application de cette spécification.

La spécification de OSGP-AES-128-PSK est structurée selon l'ordre inverse des couches logiques. Plus précisément, le reste de la présente Annexe est structuré autour des articles de premier niveau suivants:

- Contexte: cet article prépare le terrain pour la suite de sécurité en décrivant le système cible, le modèle de menace et les objectifs de conception de la suite de sécurité;

- Termes et notation: cet article décrit les termes et la notation utilisés dans cette spécification;
- Primitives cryptographiques: cet article spécifie les primitives cryptographiques de bas niveau qui servent de blocs de construction pour la suite de sécurité;
- Fonctions cryptographiques: spécifie les fonctions cryptographiques de haut niveau référencées dans la spécification du protocole;
- Clés: spécifie les clés et la hiérarchie des clés dans le système cryptographique;
- Initialisation du canal sécurisé: spécifie la méthode d'établissement d'un canal sécurisé;
- Communication par canal sécurisé: spécifie la manière d'utiliser le canal sécurisé pour protéger la communication réseau;
- Gestion des clés: spécifie la méthode de génération, de distribution et de renouvellement des clés, ainsi que les périodes de validité recommandées pour les clés;
- Messages d'erreur: spécifie les messages d'erreur courants liés à la sécurité utilisés pour indiquer les défaillances du protocole;
- Considérations de sécurité: fournit des pratiques supplémentaires recommandées pour cette suite de sécurité, ainsi que la justification sous-tendant la conception de la suite de sécurité;
- Références: répertorie les ouvrages de référence cités dans le présent document.

D.2 Contexte

D.2.1 Présentation

Le présent article fournit des informations générales sur le système cible, le modèle de menace pris en compte, les objectifs de conception et la source de cette suite de sécurité.

D.2.2 Hypothèses relatives au système

Il s'agit d'une liste d'hypothèses importantes sur le système cible.

- Le système cible est un réseau de dispositifs intégrés hérités vieux de 10 à 15 ans. Il est donc attendu que les LNAP et les NNAP soient considérablement limités en ressources, à la fois en matière de capacité de mémoire et de puissance de calcul.
- Le réseau de communication est soumis à des contraintes de bande passante, de débit, de bruit et est fortement sujet à la corruption des données lors des transferts, ce qui exige souvent des relances pour mener à bien une transaction.
- Le support de communication est partagé par un sous-réseau de nœuds (pouvant atteindre 1 024) et ne peut donc être utilisé que par un nœud à la fois. Les tentatives de communication simultanée conduisent à une corruption des paquets.
- Le protocole de transport sous-jacent ne peut pas être modifié.
- Pour résoudre ce problème de support partagé, le NNAP devient le seul initiateur de la communication. Ainsi, un NNAP est capable de contrôler l'utilisation du support dans le cadre d'une activité réseau normale.

D.2.3 Modèle de menace

Cette suite de sécurité se préoccupe d'un adversaire du réseau qui contrôle totalement le réseau de communication et connaît tout le système cryptographique, à l'exception des clés secrètes.

L'adversaire est réputé avoir accès à des ressources de calcul à haute performance.

Bien que les attaques par rejeu sur un réseau à support partagé puissent être difficiles dans la pratique, l'adversaire est réputé capable d'enregistrer, d'altérer, de supprimer et/ou d'introduire à tout moment et de manière arbitraire des messages dans le protocole.

Il importe de ne pas oublier que l'adversaire, à l'instar de tous les autres nœuds du réseau, est soumise aux mêmes contraintes imposées par le réseau à faible performance.

D.2.4 Objectifs de conception

Les objectifs de cette suite de sécurité sont les suivants:

- **Sécurité:** il convient que la conception cryptographique de la suite de sécurité soit prudente et suive les meilleures pratiques modernes et les recommandations des chiffreurs et des experts en sécurité. Il convient de définir la sécurité au moins par la confidentialité des clés de chiffrement utilisées.
- **Communication sécurisée:** il convient que la suite de sécurité assure les fonctions suivantes:
  a) **Confidentialité du message**: il convient qu'il ne soit pas possible de lire une communication sécurisée sans avoir accès à la clé (de chiffrement) utilisée.
  b) **Intégrité de l'origine (authenticité) du message**: il convient qu'il soit possible de vérifier l'origine d'un message donné sur la base d'un secret partagé.
  c) **Intégrité des données du message**: il convient qu'il soit possible de vérifier qu'un message donné n'a pas changé depuis sa création sur la base d'un secret partagé.
  d) **Prévention du rejeu de messages**: il convient qu'il soit possible de détecter et de rejeter les messages (anciens) répétés.
  e) **Prévention de la redirection de messages**: il convient qu'il soit possible de détecter et de rejeter les messages redirigés de leur destination d'origine vers une nouvelle destination.
- **Parcellisation du LNAP**: il convient que la suite de sécurité fournisse des moyens pour limiter autant que possible l'impact des compromissions (de clés) du LNAP.
- **Facile à utiliser par les développeurs**: il convient qu'il soit simple pour les développeurs de programmer une mise en œuvre sécurisée, efficace et de durée relativement courte de la suite de sécurité.
- **Adaptée pour les dispositifs hérités, à ressources limitées, sur les réseaux de communication lents**: ceci se traduit par les éléments suivants:
  f) **Empreinte minimale pour la mise en œuvre**: il convient que l'espace de mise en œuvre de la suite de sécurité soit plus petit que celui d'autres solutions de sécurité avec des objectifs de conception similaires, tout en maintenant des vitesses acceptables. Il s'agit de prendre en charge les dispositifs soumis à des contraintes de mémoire dans le système cible.
  g) **Surcharge de calcul minimale**: il convient que la suite de sécurité fonctionne à des vitesses acceptables sur les dispositifs intégrés vieux de 10 à 15 ans. En d'autres termes, il convient que la surcharge de calcul liée à l'utilisation de cette suite de sécurité ne dégrade pas les performances à des niveaux inacceptables.
  h) **Surcharge réseau minimale**: il convient que la suite de sécurité réduise la surcharge cryptographique par paquet réseau et réduise, autant que possible, le nombre de paquets nécessaires pour tenir compte de l'éventualité d'un réseau lent et peu fiable du système cible.

D.2.5 Source

Cette suite de sécurité repose en grande partie sur le protocole extensible d'authentification (EAP, Extensible Authentication Protocol) par clé prépartagée IETF RFC 5433 spécifié dans le protocole IEEE Std 802.11ac-2013. Ces deux protocoles établis partagent avec OSGP-AES-128-PSK des objectifs de conception similaires. Il est à noter que CCMP-128 est

largement considéré dans l'industrie comme étant sécurisé (il est utilisé par exemple dans la cryptographie de la Suite B de NIST à compter de 2015).

D.3 Termes, définitions, abréviations et notation

D.3.1 Termes et définitions

D.3.1.1
données associées
données authentifiées mais non chiffrées

D.3.1.2
octet
chaîne de 8 bits

D.3.1.3
CBC-MAC
chaînage de blocs de chiffrement-code d'authentification de message

Note 1 à l'article: Le terme abrégé "CBC-MAC" est dérivé du terme anglais développé correspondant "Cipher Block Chaining-Message Authentication Code".

D.3.1.4
CCM
compteur avec chaînage de blocs de chiffrement-Code d'authentification de message

Note 1 à l'article: Le terme abrégé "CCM" est dérivé du terme anglais développé correspondant "Counter with Cipher Block Chaining-Message Authentication Code".

D.3.1.5
texte chiffré
cryptogramme constitué de la charge utile chiffrée suivie d'un code d'authentification de message (MAC) protégeant à la fois l'en-tête et la charge utile. Le MAC est également utilisé pour vérifier les entrées contextuelles qui dépendent du CryptoContexte (voir D.7.2)

D.3.1.6
DC
concentrateur de données

Note 1 à l'article: Dans le modèle de réseau client-serveur, un DC est un NNAP agissant en tant que client. Dans le domaine d'application de ce profil de communication, un DC ne traite jamais les messages qui ne sont pas des réponses à sa demande.

Note 2 à l'article: Le terme abrégé "DC" est dérivé du terme anglais développé correspondant "data concentrator".

D.3.1.7
GCM
mode Galois/Counter

Note 1 à l'article: Le terme abrégé "GCM" est dérivé du terme anglais développé correspondant "Galois/Counter Mode".

D.3.1.8
LNAP
compteur intelligent

Note 1 à l'article: Il s'agit du serveur dans le modèle de réseau client-serveur. Par essence et dans le domaine d'application de ce profil de communication, un LNAP n'envoie jamais une demande au NNAP.

Note 2 à l'article: Le terme abrégé "LNAP" est dérivé du terme anglais développé correspondant "Local Network Access Point"

D.3.1.9
message
unité de données de protocole de couche adaptation (AdpPDU)

Note 1 à l'article: Il est constitué d'un en-tête suivi d'une charge utile. Lorsque la protection est exigée lors de la transmission du message, la charge utile est remplacée par un texte chiffré spécifique au message et au contexte.

D.3.1.10
en-tête
données supplémentaires placées au début d'un message authentifié mais non chiffré (partie des données associées)

D.3.1.11
charge utile
informations véritablement prévues qui sont authentifiées et chiffrées lorsque la protection est exigée

D.3.1.12
valeur unique
valeur utilisée une seule fois dans un contexte spécifié

D.3.1.13
unicast
communication entre un seul NNAP et un seul LNAP

D.3.1.14
transaction
transaction (unicast) consiste en un seul message de demande NNAP et un seul message de réponse LNAP

D.3.1.15
diffusion
communication unidirectionnelle d'un seul NNAP à un ensemble de LNAP

D.3.1.16
domaine
réseau de LNAP gérés par un NNAP

Note 1 à l'article: Un domaine ne peut être géré que par un seul NNAP à la fois.

D.3.2 Termes abrégés

Terme abrégé	Signification
BCK	Broadcast Communication Key (Clé de communication de diffusion)
DMK	Domain Master Key (Clé principale de domaine)
HSK	Handshake Key (Clé d'établissement de liaison)
MCK	Meter Commissionning Key (Clé de mise en service du compteur)
MFWK	Meter Firmware Key (Clé du microgiciel du compteur)
MMK	Meter Master Key (Clé principale de compteur)
MUK	Meter Unique Key (Clé unique du compteur)
MUKHSK	MUK-derived Handshake Key (Clé d'établissement de liaison dérivée de MUK)
MUKMCK	MUK-derived Meter Commissionning Key (Clé de mise en service du compteur dérivée de MUK)
UCK	Unicast Communication Key (Clé de communication unicast)

D.3.3 Notation

A||B	La concaténation de deux chaînes de bits A et B
0x	Le préfixe d'une chaîne de bits représentée par des caractères hexadécimaux
{...}	... est chiffré et authentifié.
"..."	... est une chaîne d'octets codée ASCII, par exemple, "ABC" est identique à 0x414243
m[:n]	Les octets n les plus à gauche de la chaîne d'octets m, par exemple, "ABC"[:2]est identique à "AB"
a.x	Le champ x dans a
UINT*n*	Entier non signé constitué de *n* bits
UINT*n*(*x..y*)	Plage de bits allant du bit d'index *x* au bit d'index *y* inclus, contenue dans UINT*n*. Le bit d'index 0 est toujours le bit le plus significatif (MSB).
FILL*n*(*x..y*)	Remplir la plage de bits allant du bit d'index *x* au bit d'index y inclus, contenue dans une chaîne de *n* bits avec des zéros. Le bit d'index 0 est toujours le bit le plus significatif (MSB).
ARRAY[*n*] OF *x*	Un tableau composé de *n* éléments de *x*.

D.3.4 Autres conventions:

- Le verbe "devoir" mis au présent de l'indicatif (doit/doivent) et sa forme verbale négative (ne doit pas/ne doivent pas) et le verbe "pouvoir" (exprimant l'autorisation) mis au présent de l'indicatif (peut/peuvent) et les expressions "exigé", "il convient", "il convient de ne pas", "recommandé" et "facultatif" utilisés dans le présent document doivent être interprétés comme décrit dans le document IETF RFC 2119.
- Les nombres entiers (types entiers) doivent être transmis dans l'ordre du réseau (c'est-à-dire en format gros boutiste).
- La numérotation à partir de zéro est utilisée pour les index et les décalages de bits et de multiplet.

D.4 Primitives cryptographiques

D.4.1 Généralités

Au niveau le plus bas (voir Tableau D.1), la suite de sécurité est construite sur trois primitives cryptographiques:

a) AES-128: variante de clé à 128 bits de l'algorithme de chiffrement par blocs symétrique approuvé par FIPS, telle que spécifiée dans FIPS PUB 197.

b) CMAC: Code d'authentification de message chiffré (CMAC) spécifié dans SP800-38B.

c) CCM: Compteur avec CBC-MAC (CCM) spécifié dans SP800-38C.

CMAC et CCM sont des algorithmes génériques qui acceptent un ensemble de paramètres. Le reste de l'Article D.4 spécifie les paramètres utilisés dans cette suite de sécurité.

Se reporter à l'Article D.13 pour la justification du choix des trois primitives cryptographiques.

NOTE Les deux applications de l'AES-128 (CCM et CMAC) utilisent uniquement l'algorithme de chiffrement de l'AES-128, et non son algorithme de déchiffrement. Aucun code n'est donc utilisé pour tout ce qui se rapporte au déchiffrement de l'AES-128, ce qui réduit considérablement le besoin en espace pour le code.

D.4.2 CMAC

CMAC est un algorithme générique de code d'authentification de message spécifié dans SP800-38B. Il possède trois paramètres principaux: un chiffrement par blocs sous-jacent (CIPH) avec une certaine taille de bloc (b) et une longueur de MAC (T_{len}). Dans cette suite de sécurité, les paramètres de CMAC sont définis comme suit:

- CIPH = AES-128; l'algorithme de chiffrement par blocs sous-jacent est AES-128 spécifié dans FIPS PUB 197.
- b = 128 (bits); la taille de bloc du CIPH est de 128 bits (il s'agit de la taille de bloc normalisée d'AES-128).
- T_{len} = 128 (bits); la longueur du MAC de sortie est de 128 bits.

Sur la base de ces paramètres, cette configuration de CBC-MAC est appelée AES-128-CMAC.

D.4.3 CCM

Compteur avec CBC-MAC (CCM) est un mode de fonctionnement générique pour le chiffrement authentifié par blocs spécifié pour la première fois dans IETF RFC 3610 et plus tard dans SP800-38C. CCM possède trois paramètres principaux: la longueur du MAC (t), la longueur en octets de la représentation binaire de la longueur en octets de la charge utile (q) et le chiffrement par blocs sous-jacent. Dans cette suite de sécurité, les paramètres de CCM sont définis comme suit:

- t = 8; le MAC est de 8 octets (64 bits).
- q = 2; le champ de longueur est de 2 octets (la longueur de la charge utile est donc limitée à 65 535 octets, ce qui est largement suffisant pour contenir la charge utile la plus grande possible).
- Block cipher = AES-128; le chiffrement par blocs sous-jacent utilisé est AES-128 comme cela est spécifié dans FIPS PUB 197.

Sur la base de ces valeurs, la longueur de la valeur unique d'entrée (n) est de 13 octets (selon la contrainte n+q=15). Dans le reste de la présente Annexe, AES-128-CCM fait référence à CCM avec la configuration ci-dessus spécifiée dans SP800-38C.

Se reporter à D.13.1.2 pour la justification du choix d'un MAC (t) de 8 octets.

NOTE Il s'agit de la même configuration de CCM qui est utilisée dans CCMP-128 spécifié dans l'IEEE Std 802.11ac-2013.

D.5 Fonctions cryptographiques

D.5.1 Présentation

Le présent article spécifie les fonctions cryptographiques utilisées dans le reste de la spécification. Ces fonctions utilisent comme blocs de construction les primitives cryptographiques spécifiées à l'Article D.4. Les fonctions cryptographiques de cette suite de sécurité sont les suivantes:

- OSGP_KDF: une fonction de dérivation de clés.
- OSGP_MAC: une fonction de code d'authentification de message.
- OSGP_MAC_VERIFY: une fonction de vérification du code d'authentification de message.
- OSGP_AE: une fonction de chiffrement authentifié.
- OSGP_AD: une fonction de déchiffrement authentifié.
- OSGP_CSPRNG: un générateur de nombres pseudo-aléatoires solide au plan cryptographique.

D.5.2 OSGP_KDF: Fonction de dérivation de clés

OSGP_KDF est une instance de la fonction générique de dérivation de clés spécifiée par SP 800-108. Cette instance utilise la construction AES-128-CMAC en tant que fonction pseudo-aléatoire sous-jacente (PRF), telle que spécifiée dans SP 800-108. Elle sert dans le système cryptographique à dériver des clés à partir d'une clé de dérivation de clés. Elle vise à rendre impossible le passage d'une clé dérivée à la clé racine. OSGP_KDF est spécifiée dans le pseudo-code ci-dessous (se reporter à SP 800-108 pour de plus amples informations).

OSGP_KDF(input_key, context, label)

Entrées:

- input_key: 16 octets de matériel de clé d'entrée.
- context (contexte): 6 octets d'informations contextuelles. Il s'agit d'un identifiant unique du compteur lors de la dérivation de clés uniques du compteur et d'un identifiant unique du domaine lors de la dérivation de clés uniques du domaine.
- label: une chaîne unique de type clé de 3 octets.

Sortie:

- 16 octets de matériel de clé de sortie.

Étapes:

1. data = 0x01 || label || 0x00 || context || 0x80
2. output_key = AES_128_CMAC(message=data, key(clé)=input_key)
3. return (retourner)output_key

Comme indiqué ci-dessus, la fonction OSGP_KDF utilise trois arguments: un matériel de clé d'entrée (input_key), des informations contextuelles (context) et un label unique de type clé (label). En interne, cette fonction combine le label et le contexte comme cela est spécifié dans SP 800-108 et utilise le résultat comme valeur d'entrée du message dans la construction AES-128-CMAC (spécifiée dans SP800-38B), tout en utilisant input_key comme matériel de clé d'entrée de 16 octets. OSGP_KDF retourne 16 octets de matériel de mise à clés dérivé pouvant être utilisé pour une nouvelle clé.

D.5.3 OSGP_MAC: Fonction de code d'authentification de message

La fonction OSGP_MAC est utilisée pour générer des codes d'authentification de message pouvant servir à vérifier l'authenticité d'un message donné à l'aide d'une clé prépartagée. Elle est fondée sur la construction AES-128-CMAC spécifiée dans SP800-38B.

La fonction OSGP_MAC est spécifiée ci-dessous.

OSGP_MAC(data, key)

Entrées:

- data (données): un message de n octets où $0 \leq n < 2^{32}$
- key (clé): une clé de16 octets

Sortie:

- code d'authentification de message (MAC) de 16 octets.

Étapes:

1. mac = AES_128_CMAC(message=data, key=key)
2. return (retourner) mac

AES_128_CMAC fait référence au processus "MAC Generation" (génération du MAC) spécifié dans SP800-38B.

D.5.4 OSGP_MAC_VERIFY: Fonction de vérification du code d'authentification de message

La fonction OSGP_MAC_VERIFY est utilisée pour vérifier un code d'authentification de message généré par OSGP_MAC. Elle est spécifiée ci-dessous.

OSGP_MAC_VERIFY(data, key, external_mac)

Entrées:

- data (données): un message de n octets où $0 \leq n < 2^{32}$
- key (clé): une clé de 16 octets
- external_mac: un code d'authentification de message de 16 octets à vérifier

Sortie:

- Soit VALID (valide) soit INVALID (INVALIDE).

Étapes:

1. internal_mac = OSGP_MAC(data=data, key=key)
2. Perform a bit-wise, constant-time comparison of internal_mac and external_mac. If they contain the same exact string of bits, then return VALID; otherwise return INVALID (effectuer une comparaison bit par bit et à temps constant de internal_mac et external_mac. S'ils contiennent la même chaîne exacte de bits, retourner alors VALID; sinon, retourner INVALID)

Comme indiqué ci-dessus, la vérification du MAC donné (external_mac) s'effectue simplement en le recalculant d'abord (internal_mac), puis en comparant les deux MAC.

Si OSGP_MAC_VERIFY indique INVALID, le message n'est donc pas authentique.

Si OSGP_MAC_VERIFY indique VALID, il convient que le message soit alors authentique selon l'assurance fournie par la construction AES-128-CMAC sous-jacente.

Il est fondamental que la comparaison bit par bit du MAC donné (external_mac) et du mac recalculé (internal_mac) soit effectuée à temps constant (étape 2). Dans le cas contraire, l'adversaire du réseau peut éventuellement déterminer le MAC (internal_mac) exact pour un message en mesurant les différences temporelles des différentes estimations de external_mac. Ceci est communément appelé attaque par canal latéral de temporisation.

D.5.5 OSGP_AE/OSGP_AD: Fonction de chiffrement/déchiffrement authentifié

D.5.5.1 Présentation

Les fonctions OSGP_AE et OSGP_AD sont utilisées respectivement pour le chiffrement authentifié et le déchiffrement authentifié.

D.5.5.2 OSGP_AE

La fonction OSGP_AE utilise le processus "Generation-Encryption" (génération-chiffrement) spécifié dans SP800-38C. Cette fonction est appelée fonction AES_128_CCM_generation_encryption dans la spécification OSGP_AE ci-dessous. Se reporter à SP800-38C pour de plus amples informations.

OSGP_AE(payload, associated_data, nonce, key)

Entrées:

- payload (charge utile): une charge utile de n octets où $0 \leq n < 2^{16}$. Cette chaîne d'octets est authentifiée et chiffrée.
- associated_data: données associées de a octet où $0 \leq a < 30$. Cette chaîne d'octets est authentifiée mais non chiffrée.
- nonce (valeur unique): une valeur unique de 13 octets.
- key (clé): une clé de 16 octets

Sortie:

- $n + t$ octets de texte chiffré où n est le nombre d'octets de la charge utile d'entrée et t est la longueur du MAC spécifiée en D.4.3.

Étapes:

1. ciphertext (texte chiffré) = AES_128_CCM_generation_encryption(
   P=payload,
   N=nonce,
   A=associated_data,
   K=key)
2. return (retourner) ciphertext

Selon les spécifications de SP800-38C, le texte chiffré est la charge utile chiffrée immédiatement suivie du MAC de t octets de longueur (t est spécifié en D.4.3). Le MAC est calculé sur la base des données associées et de la charge utile.

Selon les exigences du CCM, la valeur unique ne doit jamais se répéter sous la même clé. Ceci est assuré par les schémas de génération de la valeur unique spécifiés en D.8.3.3 et D.8.4.3.

D.5.5.3 OSGP_AD

La fonction OSGP_AD utilise le processus "Decryption-Verification" (déchiffrement-vérification) spécifié dans SP800-38C. Cette fonction est appelée fonction AES_128_CCM_decryption_verification dans la spécification OSGP_AD ci-dessous. Se reporter à SP800-38C pour de plus amples informations.

OSGP_AD(ciphertext, associated_data, nonce, key)

Entrées:

- ciphertext (texte chiffré): texte chiffré présumé de c octets où $t \leq c < 2^{16} + t$ et t est spécifié en D.4.3.
- associated_data: données associées (voir la spécification OSGP_AE pour de plus amples informations).
- nonce: valeur unique (voir la spécification OSGP_AE pour de plus amples informations).
- key (clé): une clé de 16 octets

Sorties:

- status (état): VALID ou INVALID.
- payload (charge utile): si état est VALID, alors sortie de charge utile de n octets de OSGP_AE; sinon (état est INVALID), sortie de zéro octet.

Étapes:

1. payload_or_invalid = AES_128_CCM_decryption_verification(
   C=ciphertext,
   A=associated_data,
   N=nonce,
   K=key)
2. if (si) payload_or_invalid == INVALID:
   status = INVALID
   payload = ""
   else (ou bien):
   status = VALID
   payload = payload_or_invalid
3. return (retourner) status, payload

Lors de la mise en œuvre de la fonction AES_128_CCM_decryption_verification (c'est-à-dire du processus de déchiffrement-vérification spécifié dans SP800-38C), il est nécessaire de prêter attention à la comparaison des deux MAC à temps constant (point 10 de 6.2, paragraphe Étapes de SP800-38C).

D.5.6 OSGP_CSPRNG: Générateur de nombres pseudo-aléatoires sécurisé par cryptographie

La fonction OSGP_CSPRNG est utilisée pour générer des nombres aléatoires adaptés à des fins cryptographiques.

Le présent document ne définit pas de manière spécifique un générateur (sous-jacent) de nombres pseudo-aléatoires sécurisé par cryptographie (CSPRNG). Il revient donc à l'implémenteur de choisir un CSPRNG adéquat.

Dans la spécification de OSGP_CSPRNG ci-dessous, le CSPRNG sous-jacent est désigné par la fonction "platform_csprng (n)" qui génère n octets aléatoires adaptés à une utilisation cryptographique.

OSGP_CSPRNG(num_of_bytes)

Entrée:

- num_of_bytes: nombre d'octets aléatoires à générer.

Sortie:

- Octets aléatoires (num_of_bytes) adaptés à des fins cryptographiques.

Étapes:

1. If (Si) num_of_bytes <= 0:
   return (retourner) ""
2. random_bytes = platform_csprng(num_of_bytes)
3. return (retourner) random_bytes

D.6 Clés

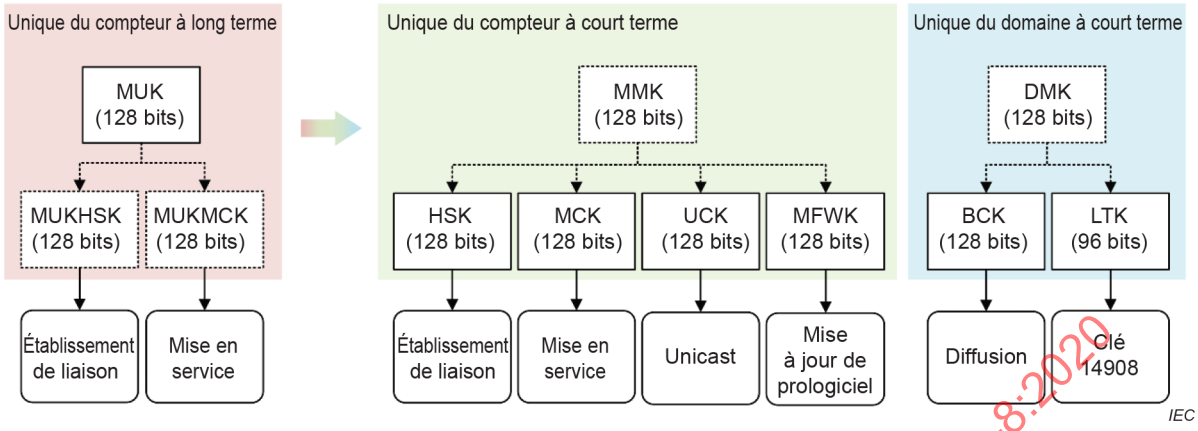

Figure D.1 – Diagramme de la hiérarchie des clés

La Figure D.1 ci-dessus donne une vue d'ensemble des clés utilisées dans le système cryptographique. Chaque case carrée représente une clé avec son identifiant unique et sa longueur en bits. Une flèche en pointillé indique une dérivation de clé à sens unique. Par exemple, la clé UCK est dérivée de la clé MMK de manière à rendre le passage de l'UCK à la MMK impossible. La ligne en pointillé autour de la MMK indique que la MMK est une clé temporaire de dérivation de clés. Il convient qu'à la fois le NNAP et le LNAP l'effacent de manière sécurisée après l'avoir utilisée pour dériver des sous-clés. Il en va de même pour la DMK, sauf qu'il convient que seul le LNAP l'efface de manière sécurisée après avoir dérivé ses sous-clés, et non le NNAP (celui-ci en a besoin pour la mise en service de nouveaux LNAP), comme cela est spécifié à l'Article D.7.

De même, les lignes en pointillés autour des sous-clés MUK indiquent qu'il convient que ces sous-clés soient effacées après première utilisation (comme cela est spécifié à l'Article D.7). Un carré arrondi indique un processus. Par exemple, "Commission" (mis en service) représente le processus de mise en service spécifié à l'Article D.7. Une flèche pleine reliant une clé à un processus signifie que la clé est utilisée pour sécuriser ce processus spécifique. Par exemple, la clé HSK protège la transaction d'établissement de liaison lors de l'initialisation du canal sécurisé spécifiée à l'Article D.7.

Dans le diagramme, les clés sont regroupées en fonction de leur période de validité et de leur domaine d'application. Les groupes rouge, vert et bleu sont constitués respectivement de clés uniques de compteur à long terme, de clés uniques de compteur à court terme et de clés uniques de domaine à court terme.

Il existe une exception: il convient que les clés à long terme dérivées de MUK soient effacées après utilisation.

Les clés à long terme sont utilisées pour (r-)établir la confiance initiale entre un LNAP et un NNAP. Dès lors que la confiance initiale est établie, elle sert à établir de manière sécurisée les clés à court terme, qui sont donc utilisées à partir de cet instant (ceci est indiqué par la flèche allant des clés à long terme aux clés à court terme). Si les clés à court terme sont considérées comme étant compromises ou si, pour une raison quelconque, elles ne sont plus accessibles, les clés à long terme peuvent donc être utilisées.

Voir l'Article D.10 pour de plus amples informations sur la gestion des clés.

Voir le Tableau D.2 pour une définition détaillée de chaque clé (taille exprimée en bits)

Tableau D.2 – Type de clés et domaine d'application

	Taille	Origine	Transférée	But	Période de validité	Domaine d'applicatior	Accès
MUK	128	CSPRNG	Hors du domaine d'application de cette spécification	Dérivation de clés (Article D.7)	Long terme	Unique du compteur	LNAP, NNAP
MUKHSK	128	Dérivée de MUK	Non	Établissement de liaison (Article D.7)	Utilisée une seule fois, puis effacée	Unique du compteur	LNAP, NNAP
MUKMCK	128	Dérivée de MUK	Non	Mise en service (Article D.7)	Utilisée une seule fois, puis effacée	Unique du compteur	LNAP, NNAP
MMK	128	CSPRNG	Demande de mise en service	Dérivation de clés (Article D.7)	Effacée après dérivation de ses sous-clés	Unique du compteur	LNAP, NNAP
HSK	128	Dérivée de MMK	Non	Établissement de liaison (Article D.7)	Court terme	Unique du compteur	LNAP, NNAP
MCK	128	Dérivée de MMK	Non	Mise en service (Article D.7)	Court terme	Unique du compteur	LNAP, NNAP
UCK	128	Dérivée de MMK	Non	Communication unicast (Article D.8)	Court terme	Unique du compteur	LNAP, NNAP
MFWK	128	Dérivée de MMK	Non	Mise à jour de micrologiel (Article D.9)	Court terme	Unique du compteur	LNAP, NNAP
DMK	128	CSPRNG	Demande de mise en service	Dérivation de clés (Article D.7)	Effacée après dérivation de ses sous-clés	Unique du domaine	Domaine, NNAP
LTK	96	Dérivée de DMK	Non	Authentification ISO/IEC 14908-1	Court terme	Unique du domaine	Compteur, NNAP
BCK	128	Dérivée de DMK	Non	Communication de diffusion (Article D.8)	Court terme	Unique du domaine	Domaine, NNAP

Un accès "LNAP, NNAP" signifie que la clé particulière est partagée entre un LNAP et un NNAP, tandis qu'un accès "Domaine, NNAP" signifie que la clé est partagée entre un NNAP et tous les LNAP de son domaine spécifique.

Comme indiqué ci-dessus, un générateur de nombres pseudo-aléatoires solide au plan cryptographique (CSPRNG) adapté à la génération de clés cryptographiques doit être utilisé pour générer les clés suivantes:

- MUK
- MMK
- DMK

Les autres clés du système cryptographique sont des clés dérivées de MUK, MMK ou DMK.

La manière dont la MUK est prépartagée en toute sécurité entre le LNAP et le NNAP ne relève pas du domaine d'application de cette spécification.

D.7 Initialisation du canal sécurisé

D.7.1 Présentation

Avant de pouvoir communiquer en toute sécurité sur le réseau, il est nécessaire que le NNAP établisse un contexte cryptographique (CryptoContexte) avec l'ensemble des LNAP qu'il gère sur son domaine. Un CryptoContexte est établi par LNAP à l'aide de clés uniques de compteur prépartagées, destinées à l'authentification et au chiffrement.

Le présent article spécifie ce qu'est un CryptoContexte et la manière dont il est établi entre un NNAP et un LNAP.

D.7.2 État du canal sécurisé (CryptoContexte)

Un CryptoContexte contient les champs suivants:

- HSK: clé d'établissement de liaison, voir l'Article D.6;
- MCK: clé de mise en service du compteur, voir l'Article D.6;
- UCK: clé de communication unicast, voir l'Article D.6;
- MFWK: clé du microgiciel du compteur, voir l'Article D.6;
- BCK: clé de communication de diffusion, voir l'Article D.6;
- LTK: clé ISO/IEC 14908-1, voir l'Article D.6;
- USeq: numéro de séquence de demande de 4 octets utilisé pour la communication unicast;
- BSeq: numéro de séquence de diffusion unique au domaine de 4 octets utilisé pour la communication de diffusion;
- URand: 2 octets générés aléatoirement;
- BRand: 5 octets générés aléatoirement;
- DomainID: domain_ID (ID de domaine) de réseau de 3 octets;
- SubnetID: subnet_ID (ID de sous-réseau) de réseau de 1 octet;
- NodeID: node_ID (ID de nœud) de réseau de 1 octet;
- MID: ID unique de compteur de 6 octets.

Tous les champs à l'exception de USeq, BSeq, URand, BRand, DomainID, SubnetID, NodeID et MID constituent de l'information secrète qui doit être protégée de tout accès non autorisé et effacée de manière sécurisée lorsqu'elle n'est plus nécessaire. L'URand et le BRand ne doivent pas être traités comme de l'information secrète bien qu'ils ne soient jamais transmis en clair. Voir D.8.3 et D.8.4 pour des informations sur les utilisations respectives de l'URand et du BRand.

Le MID est réputé être connu du NNAP et du LNAP avant la première initialisation. La manière dont le MID est prépartagé avant la première initialisation ne relève pas du domaine d'application de cette spécification. Il est à noter que le MID n'est pas un secret.

Un CryptoContexte reste valide jusqu'à la prochaine initialisation. Lorsqu'un nouveau CryptoContexte est établi, toutes les informations sensibles de l'ancien CryptoContexte doivent être effacées de manière sécurisée et remplacées par de nouvelles valeurs.

D.7.3 Flux

Le présent paragraphe spécifie la manière dont un CryptoContexte utilisé pour le canal sécurisé est (ré)initialisé entre un NNAP et un LNAP. Une vue d'ensemble du flux de messages est donnée à la Figure D.2 (voir l'Article D.2).

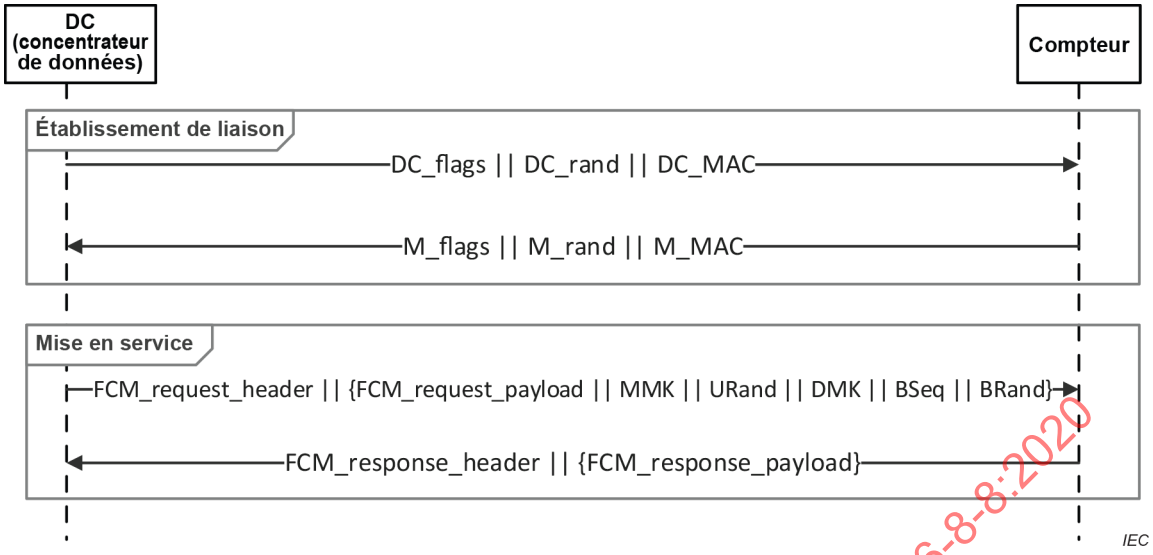

Figure D.2 – Établissement d'un canal sécurisé

L'achèvement du processus nécessite deux transactions (une transaction comprend 1 message de demande et 1 message de réponse, soit 4 messages au total). La première transaction est appelée transaction d'"établissement de liaison". Elle consiste en un flux "défi-réponse" qui rafraîchit et met en place le processus d'authentification mutuelle. La deuxième transaction est appelée transaction de "mise en service" qui configure le LNAP et (ré)initialise le CryptoContexte partagé entre le NNAP et le LNAP. Les deux transactions sont liées cryptographiquement l'une à l'autre, ce qui rend impossible le fractionnement, la combinaison ou la retransmission des anciens flux sans que cela ne soit détecté par le NNAP et/ou le LNAP.

Les formats des différents types de messages sont spécifiés en D.7.7. Les messages d'erreur sont spécifiés à l'Article D.12.

La spécification du processus se fait à l'aide des fonctions cryptographiques définies à l'Article D.5. Elles sont préfixées par OSGP_.

L'ensemble du processus est spécifié ci-dessous.

Soit:

- handshake_key: la clé d'établissement de liaison utilisée pour authentifier les messages d'établissement de liaison. Il s'agit de la HSK du CryptoContexte actuel ou de la MUKHSK. La MUKHSK est dérivée comme suit:
  OSGP_KDF(key=MUK, context=MID, label="hsk");
- commission_request_key: la clé utilisée pour protéger la demande de mise en service (CommissionRequest). Il s'agit de la MCK du CryptoContexte actuel ou de la MUKMCK. La MUKMCK est dérivée comme suit:
  OSGP_KDF(key=MUK, context=MID, label="mck");
- commission_response_key: la clé utilisée pour protéger la réponse de mise en service (CommissionResponse). Il s'agit de la nouvelle MCK, où *nouvelle* indique qu'il s'agit d'une clé dérivée de la nouvelle MMK présente dans la CommissionRequest;
- MID: l'identifiant unique du compteur pour le LNAP concerné;
- BSeq: la valeur actuelle du numéro de séquence de diffusion du NNAP;
- BRand: les données aléatoires actuelles pour une diffusion sécurisée (uniquement pour le NNAP);
- DMK: la DMK actuelle du domaine (uniquement pour le NNAP).

Résultat d'un processus réussi:

- Un CryptoContexte unique de compteur partagé entre le NNAP et le LNAP, qui peut être utilisé pour une communication sécurisée spécifiée à l'Article D.8.

Étapes:

Transaction d'établissement de liaison

1) Le NNAP construit un message ChallengeRequest (demande de défi):
2) DC_flags = 0x061001
3) DC_rand = OSGP_CSPRNG(16)
4) DC_MAC = OSGP_MAC(data=MID || DC_flags || DC_rand,
   key=handshake_key)
5) Le NNAP envoie le ChallengeRequest au LNAP conformément à la Figure D.3:

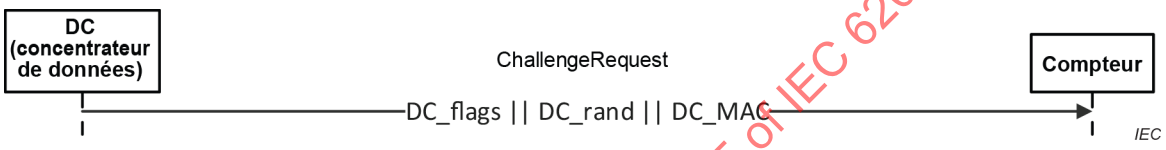

Figure D.3 – Demande de défi

6) Le LNAP reçoit le ChallengeRequest
7) Le LNAP vérifie DC_MAC
   status = OSGP_MAC_VERIFY(data=MID || DC_flags || DC_rand,
   key=handshake_key,
   external_mac=DC_MAC)
8) Si "status" (état) est INVALID, le LNAP répond par un AuthenticationFailure (échec d'authentification) et abandonne le processus; dans le cas contraire, le LNAP poursuit le processus.
9) Le NNAP construit un message ChallengeResponse (réponse à la demande de défi):
10) M_flags = 0x0600ssoo (ss est la valeur des modes de sécurité pris en charge et oo le masque de bits des fonctions facultatives, voir D.7.7.3 pour de plus amples informations).
11) M_rand = OSGP_CSPRNG(16)
12) M_MAC = OSGP_MAC(data=MID || M_flags || M_rand || DC_rand,
    key=handshake_key)
13) Le LNAP envoie le ChallengeResponse au NNAP conformément à la Figure D.4:

Figure D.4 – Réponse à la demande de défi

14) Le NNAP reçoit le ChallengeResponse

15) Le NNAP vérifie M_MAC:
status = OSGP_MAC_VERIFY(data=MID || M_flags || M_rand || DC_rand,
key=handshake_key,
external_mac=M_MAC)

16) Si "status" (état) est INVALID, il convient que le NNAP enregistre l'événement et il doit abandonner le processus; dans le cas contraire, le NNAP poursuit le processus.

Transaction de mise en service

17) Le NNAP construit un message CommissionRequest (demande de mise en service):

18) FCM_request_header = voir D.7.7.4

19) FCM_request_payload = voir D.7.7.4

20) MMK = OSGP_CSPRNG(16)

21) URand = OSGP_CSPRNG(2)

22) Le NNAP construit une "nonce" (valeur unique) de demande de 13 octets: nonce = 0x02 || DC_rand[:6] || M_rand[:6]

23) Le NNAP construit les données associées pour le message CommissionRequest: associated_data = M_mac || FCM_request_header

24) Le NNAP effectue un chiffrement authentifié du CommissionRequest: ciphertext = OSGP_AE(
payload=FCM_request_payload || MMK || URand || DMK || BSeq || BRand,
associated_data=associated_data,
nonce=nonce,
key=commission_request_key)

25) Le NNAP envoie FCM_request_header || ciphertext (unmessage CommissionRequest protégé) au LNAP conformément à la Figure D.5:

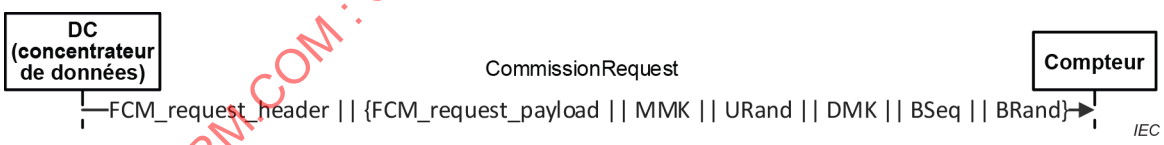

Figure D.5 – Demande de mise en service

26) Le LNAP reçoit le CommissionRequest protégé (header || ciphertext)

27) Le LNAP construit la valeur unique de demande comme le fait le NNAP à l'étape 10:
nonce = 0x02 || DC_rand[:6] || M_rand[:6]

28) Le LNAP construit les données associées comme le fait le NNAP à l'étape 11: associated_data = M_mac || FCM_request_header

29) Le LNAP effectue un déchiffrement authentifié du CommissionRequest protégé: status, payload = OSGP_AD(ciphertext=ciphertext,
associated_data=associated_data,
nonce=nonce,
key=commission_request_key)

30) Si "status" (état) est INVALID, le LNAP répond par un message AuthenticationFailure (échec d'authentification) et abandonne le processus; dans le cas contraire, le LNAP poursuit le processus (la charge utile est le texte chiffré déchiffré).

31) LNAP met à jour son CryptoContexte interne en fonction des valeurs que contient le CommissionRequest authentifié et déchiffré:

a. CryptoContext.HSK = OSGP_KDF(
input_key=CommissionRequest.MMK,
context=MID,
label="hsk")
b. CryptoContext.MCK = OSGP_KDF(
input_key=CommissionRequest.MMK,
context=MID,
label="mck")
c. CryptoContext.UCK = OSGP_KDF(
input_key=CommissionRequest.MMK,
context=MID,
label="uck")
d. CryptoContext.MFWK = OSGP_KDF(
input_key=CommissionRequest.MMK,
context=MID,
label="fwk")
e. CryptoContext.BCK = OSGP_KDF(
input_key=CommissionRequest.DMK,
context=CommissionRequest.DomainID || 0x000000,
label="bck")
f. CryptoContext.LTK = OSGP_KDF(
input_key=CommissionRequest.DMK,
context=CommissionRequest.DomainID || 0x000000,
label="ltk")[:12]
g. CryptoContext.USeq = 1
h. CryptoContext.URand = CommissionRequest.URand
i. CryptoContext.BRand = CommissionRequest.BRand
j. CryptoContext.BSeq = CommissionRequest.BSeq
k. CryptoContext.DomainID = CommissionRequest.DomainID
l. CryptoContext.SubnetID = CommissionRequest.SubnetID
m. CryptoContext.NodeID = CommissionRequest.NodeID

32) Le LNAP veille à ce que les valeurs écrasées de son CryptoContexte soient effacées de manière sécurisée.

33) Le LNAP efface de manière sécurisée la MMK et la DMK.

34) Le LNAP met à jour sa configuration comme cela est spécifié en D.7.5.

35) Le LNAP construit une "nonce" (valeur unique) de réponse de 13 octets: nonce = 0x03 || DC_rand[:6] || M_rand[:6]

36) Le LNAP construit les données associées pour la réponse:
associated_data =
nonce ||
CryptoContext.DomainID || CryptoContext.SubnetID || CryptoContext.NodeID || fcm_response_header

37) Le LNAP construit et effectue un chiffrement authentifié du CommissionResponse:
ciphertext = OSGP_AE(payload=fcm_response_payload,
associated_data=associated_data,
nonce=nonce,
key=commission_response_key)

38) Le LNAP envoie FCM_response_header || ciphertext (un message CommissionResponse protégé) au LNAP conformément à la Figure D.6:

Figure D.6 – Réponse de mise en service

39) Le NNAP reçoit un CommissionResponse protégé

40) Le NNAP construit la "nonce" (valeur unique) de réponse de 13 octets: nonce = 0x03 || DC_rand[:6] || M_rand[:6]

41) Le NNAP construit les données associées comme le fait le LNAP à l'étape 37: associated_data =
nonce ||
CryptoContext.DomainID || CryptoContext.SubnetID || CryptoContext.NodeID || FCM_response_header

42) Le NNAP effectue un déchiffrement authentifié du CommissionResponse: payload, status=OSGP_AD(ciphertext=ciphertext,
associated_data=associated_data,
nonce=nonce,
key=commission_response_key)

43) Si "status" (état) est INVALID, il convient que le NNAP enregistre l'événement et il doit abandonner le processus; dans le cas contraire, le NNAP poursuit le processus.

44) Le NNAP met à jour le CryptoContexte par LNAP exactement comme le LNAP (étape 24).

45) Le NNAP efface de manière sécurisée la MMK.

46) Il convient que le NNAP enregistre l'événement d'un processus réussi.

Noter que dans certains déploiements, il peut être préférable d'utiliser la MUK pour toutes les demandes de mise en service. Toutefois, pour les systèmes dans lesquels la durée de vie de la clé MUK est très longue ou dans lesquels cette clé ne peut pas être mise à jour, l'utilisation de la MUK pour des mises à jour fréquentes de clés peut s'avérer peu souhaitable. En fin de compte, il s'agit d'une décision qu'il convient que l'utilisateur prenne. Ce protocole permet simplement aux différentes approches de répondre à différents besoins.

Comme spécifié ci-dessus, la MMK doit être fraîchement générée à chaque initialisation. Cette génération assure l'indépendance des clés et leur unicité pratique et constitue une partie importante des schémas de numéros de séquence. Les numéros de séquence de monodiffusion (USeq) doivent être réinitialisés à 1 lors de la mise en service. Sinon, la réutilisation d'une MMK entraînerait la réutilisation d'un ou de plusieurs numéros de séquence, une réutilisation qui en retour fausse les hypothèses selon lesquelles les numéros de séquence sont en augmentation. L'augmentation des numéros de séquence est une propriété sur laquelle se reposent fondamentalement la valeur unique et les schémas de prévention du rejeu.

S'il n'est pas souhaitable d'utiliser la clé MUK à long terme, le NNAP doit conserver une trace du CryptoContexte actuel et du nouveau jusqu'à ce qu'il reçoive une réponse valide à sa requête CommissionRequest. Une réponse valide est un message CommissionResponse. La non-réception d'un CommissionResponse n'est pas une réponse valide. Cela permet d'assurer que le NNAP est capable de se réinitialiser sans avoir à revenir à la MUK à long terme. Par exemple, si le message CommissionResponse est perdu pendant le transfert et n'atteint jamais le NNAP, le NNAP dispose de deux options: effectuer une nouvelle (fraîche) réinitialisation en utilisant le CryptoContexte actuel (à nouveau) ou essayer de faire de même, mais avec le nouveau CryptoContexte. Si une option échoue, il convient que le NNAP essaie l'autre. Si les deux options échouent, le NNAP peut tenter à nouveau ou finalement utiliser la MUK à long terme.

Il convient que le LNAP attende le message CommissionRequest au plus pendant n secondes depuis la réception du message ChallengeRequest. Il convient que le LNAP abandonne le processus après l'écoulement des n secondes sans une réception de CommissionRequest. Il incombe à la spécification d'accompagnement de décider de la valeur de n car elle est spécifique à chaque système.

Il convient également que le LNAP limite le nombre de messages ChallengeRequest qu'il accepte au cours d'une période donnée. Par exemple, lorsqu'un LNAP a reçu 100 demandes de défi en une heure, il convient qu'il attende une heure avant d'autoriser 100 autres demandes de défi. Il convient de veiller à ce que l'exploitation de ce mécanisme ne soit pas triviale pour l'adversaire du réseau.

En cas d'échec de l'authentification, il convient que le NNAP enregistre un événement de détection d'intrusion que le système centralisé de gestion et de télérelevé peut analyser et auquel il peut apporter une réponse aux incidents appropriée. Par exemple, le système centralisé de gestion et de télérelevé peut décider d'utiliser la MUK pour réinitialiser le LNAP si les clés à court terme ne fonctionnent pas. Toutefois, cela amènerait le NNAP à déclencher un événement incident.

D.7.4 Négociation de la suite de sécurité

Le NNAP spécifie la suite de sécurité souhaitée (dans le champ "Security Mode" (mode de sécurité) de l'en-tête de demande FCM non chiffré, voir Tableau D.5). Si le LNAP ne prend pas en charge le mode de sécurité souhaité, il produit un accusé de réception négatif (NACK) et indique dans la réponse le ou les modes de sécurité pris en charge. Le NNAP peut ensuite décider de poursuivre l'une des alternatives ou de signaler une incompatibilité de sécurité. Il convient de prendre des précautions pour éviter les attaques de rétrogradation, car les NACK peuvent être forgés par l'adversaire du réseau. Il est recommandé de configurer si possible le NNAP pour qu'il n'utilise que la suite OSGP-AES-128-PSK et qu'il signale un événement de sécurité si le LNAP (attaquant) répond avec une liste inattendue de modes de sécurité pris en charge.

D.7.5 Mise en service du LNAP

Outre l'échange d'informations de sécurité, la mise en service configure le LNAP pour une communication normale et révèle les capacités du LNAP. Ces informations sont regroupées dans une seule transaction de message (CommissionRequest et CommissionResponse) pour accélérer la mise en service.

Les champs suivants des messages de mise en service sont nécessaires pour l'adressage et la temporisation de message ISO/IEC 14908-1:

- domain_ID,
- subnet_ID,
- node_ID,
- non-group RX Timer.

Le lecteur doit garder à l'esprit que les identifiants domain_ID, subnet_ID et node_ID du LNAP sont également liés au CryptoContexte en cours.

Les champs suivants permettent d'identifier les capacités du LNAP:

- Optional Features (fonctionnalités facultatives),
- Compatibility Set (réglage de compatibilité)
- Program ID (identifiant de programme),
- Inactive Phases (phases inactives),
- Hardware configuration Length (longueur de configuration du matériel),
- Hardware configuration record (enregistrement de la configuration du matériel),
- LNAP Model (modèle de compteur).

D.7.6 Traitement des erreurs et détection des intrusions

La gestion des erreurs lors de l'initialisation du canal sécurisé est simple: tout type d'état d'erreur est fatal et conduit forcément à l'abandon de l'ensemble du processus. L'abandon du processus invalide également les deux transactions, ce qui signifie qu'aucune des informations transférées ne peut être digne de confiance. Par conséquent, le NNAP doit lancer une *nouvelle* initialisation *s*'il décide d'essayer à nouveau.

Si un AuthenticationFailure est renvoyé, il convient de l'enregistrer et le transmettre à un système de détection d'intrusion pour une réponse aux incidents. En pratique, il convient qu'il n'y ait aucun cas légitime de message AuthenticationFailure: un attaquant a injecté le message d'erreur ou bien la clé a été remplacée par quelque chose d'inattendu. Dans les deux cas, il s'agit d'un état anormal qu'il convient de signaler.

D.7.7 Messages

D.7.7.1 Présentation

Le Paragraphe D.7.7 spécifie les quatre messages de non-erreur utilisés lors de l'initialisation du canal sécurisé.

D.7.7.2 ChallengeRequest

Le Tableau D.3 présente la structure de la demande de défi envoyée par le NNAP.

Tableau D.3 – Demande de défi

	Décalage (octet)	Type	Nom	A	E	Description
DC_ flags	octet 0	unsigned	ISO/IEC 14908-1 Code(Code ISO/IEC 14908-1)	Oui	Non	0x06
	octet 1	unsigned	Configuration	Oui	Non	Table de bits b0 – b2: Sous code FCM. 0 b3 – b5: Mode de sécurité. 2=AES-128 b6 – b7: réservé
	octet 2	unsigned	Optional Features (fonctionnalités facultatives)	Oui	Non	Bit 0: Si défini, le NNAP prend en charge la MCK Bits 1-7: Réservé
	octet 3	octet-string(SIZE(16))	DC_rand	Oui	Non	Valeur de défi aléatoire générée par le NNAP
	octet 19	octet-string(SIZE(16))	DC_MAC	MAC	Non	Le MAC couvrant les données précédentes