

# **SLOVENSKI STANDARD**

## **SIST-TS CLC/TS 50136-9:2017**

**01-oktober-2017**

**Nadomešča:**

**SIST-TS CLC/TS 50136-9:2014**

---

**Alarmni sistemi - Sistemi in oprema za prenos alarma – 9. del: Zahteve za skupni protokol za prenos alarma po internetnem protokolu**

Alarm systems - Alarm transmission systems and equipment - Part 9: Requirements for common protocol for alarm transmission using the Internet Protocol (IP)

Alarmanlagen - Alarmübertragungsanlagen und -einrichtungen - Teil 9: Anforderungen an standardisierte Protokolle zur Alarmübertragung unter Nutzung des Internetprotokolls (IP)

[SIST-TS CLC/TS 50136-9:2017](http://standards.iteh.ai/SIST-TS-CLC/TS-50136-9-2017)

Systemes d'alarmes - Systemes et équipements de transmission d'alarme - Partie 9 : Exigences pour le protocole commun de transmission d'alarme utilisant le protocole Internet (IP)

**Ta slovenski standard je istoveten z: CLC/TS 50136-9:2017**

---

**ICS:**

|           |                                   |                             |
|-----------|-----------------------------------|-----------------------------|
| 13.320    | Alarmni in opozorilni sistemi     | Alarm and warning systems   |
| 33.040.40 | Podatkovna komunikacijska omrežja | Data communication networks |

**SIST-TS CLC/TS 50136-9:2017**

**en**

## **iTeh STANDARD PREVIEW** **(standards.iteh.ai)**

[SIST-TS CLC/TS 50136-9:2017](https://standards.iteh.ai/catalog/standards/sist/b97f1291-3021-4bb9-b46f-3a69d17a558b/sist-ts-clc-ts-50136-9-2017)

<https://standards.iteh.ai/catalog/standards/sist/b97f1291-3021-4bb9-b46f-3a69d17a558b/sist-ts-clc-ts-50136-9-2017>

TECHNICAL SPECIFICATION  
SPÉCIFICATION TECHNIQUE  
TECHNISCHE SPEZIFIKATION

**CLC/TS 50136-9**

August 2017

ICS 13.320; 33.040.40

Supersedes CLC/TS 50136-9:2013

English Version

**Alarm systems - Alarm transmission systems and equipment -  
Part 9: Requirements for common protocol for alarm  
transmission using the Internet Protocol (IP)**

Systèmes d'alarmes - Systèmes et équipements de  
transmission d'alarme - Partie 9 : Exigences pour le  
protocole commun de transmission d'alarme utilisant le  
protocole Internet (IP)

Alarmanlagen - Alarmübertragungsanlagen und -  
einrichtungen - Teil 9: Anforderungen an standardisierte  
Protokolle zur Alarmübertragung unter Nutzung des  
Internetprotokolls (IP)

This Technical Specification was approved by CENELEC on 2017-05-29.

CENELEC members are required to announce the existence of this TS in the same way as for an EN and to make the TS available promptly at national level in an appropriate form. It is permissible to keep conflicting national standards in force.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

[SIST-TS CLC/TS 50136-9:2017](https://standards.iteh.ai/catalog/standards/sist/b97f1291-3021-4bb9-b46f-3a69d17a558b/sist-ts-clc-ts-50136-9-2017)

<https://standards.iteh.ai/catalog/standards/sist/b97f1291-3021-4bb9-b46f-3a69d17a558b/sist-ts-clc-ts-50136-9-2017>



European Committee for Electrotechnical Standardization  
Comité Européen de Normalisation Electrotechnique  
Europäisches Komitee für Elektrotechnische Normung

**CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels**

# Contents

Page

|                                                     |           |
|-----------------------------------------------------|-----------|
| <b>European foreword</b> .....                      | <b>6</b>  |
| <b>1 Scope</b> .....                                | <b>7</b>  |
| <b>2 Normative references</b> .....                 | <b>7</b>  |
| <b>3 Terms, definitions and abbreviations</b> ..... | <b>7</b>  |
| 3.1 Terms and definitions .....                     | 7         |
| 3.2 Abbreviations.....                              | 7         |
| <b>4 Objective</b> .....                            | <b>8</b>  |
| <b>5 Messaging</b> .....                            | <b>8</b>  |
| 5.1 General .....                                   | 8         |
| 5.2 Message format overview.....                    | 9         |
| 5.2.1 General.....                                  | 9         |
| 5.2.2 Identifiers.....                              | 9         |
| 5.2.3 Message format.....                           | 10        |
| 5.2.4 Connection handle.....                        | 11        |
| 5.2.5 Device ID.....                                | 11        |
| 5.2.6 Message ID .....                              | 12        |
| 5.2.7 Message Length .....                          | 13        |
| 5.2.8 Sequence numbers.....                         | 13        |
| 5.2.9 Flags .....                                   | 13        |
| 5.3 Padding and message length.....                 | 13        |
| 5.3.1 General.....                                  | 13        |
| 5.3.2 Message Length.....                           | 14        |
| 5.4 Hashing.....                                    | 14        |
| 5.4.1 General.....                                  | 14        |
| 5.4.2 Invalid hash - transmitter response.....      | 14        |
| 5.4.3 Invalid hash - receiver response.....         | 14        |
| 5.5 Encryption .....                                | 14        |
| 5.5.1 General.....                                  | 14        |
| 5.5.2 Key exchange.....                             | 15        |
| 5.6 Timeouts and retries.....                       | 15        |
| 5.7 Version number .....                            | 16        |
| 5.8 Reverse commands.....                           | 16        |
| 5.9 Initial values.....                             | 16        |
| <b>6 Message types</b> .....                        | <b>17</b> |
| 6.1 Path supervision .....                          | 17        |
| 6.1.1 General.....                                  | 17        |
| 6.1.2 Poll message .....                            | 17        |
| 6.1.3 Poll response.....                            | 18        |
| 6.2 Event message format.....                       | 18        |
| 6.2.1 General.....                                  | 18        |
| 6.2.2 Event field.....                              | 20        |
| 6.2.3 Time event field .....                        | 20        |
| 6.2.4 Time message field.....                       | 20        |
| 6.2.5 Link field – IP Address .....                 | 21        |
| 6.2.6 Link field – Port number.....                 | 21        |
| 6.2.7 Link field – URL .....                        | 21        |
| 6.2.8 Link field - Filename.....                    | 22        |
| 6.2.9 Alarm Text.....                               | 22        |
| 6.2.10 Site Name.....                               | 22        |
| 6.2.11 Building Name .....                          | 22        |
| 6.2.12 Location.....                                | 22        |
| 6.2.13 Room.....                                    | 23        |

|          |                                                                      |           |
|----------|----------------------------------------------------------------------|-----------|
| 6.2.14   | Alarm Trigger.....                                                   | 23        |
| 6.2.15   | Longitude.....                                                       | 23        |
| 6.2.16   | Latitude.....                                                        | 23        |
| 6.2.17   | Altitude .....                                                       | 24        |
| 6.3      | Event response format .....                                          | 24        |
| 6.4      | Configuration messages.....                                          | 25        |
| 6.4.1    | General.....                                                         | 25        |
| 6.4.2    | Connection handle request.....                                       | 25        |
| 6.4.3    | Connection handle response .....                                     | 25        |
| 6.4.4    | Device ID request.....                                               | 26        |
| 6.4.5    | Device ID response .....                                             | 27        |
| 6.4.6    | Encryption selection request.....                                    | 27        |
| 6.4.7    | Encryption selection response .....                                  | 28        |
| 6.4.8    | Encryption key exchange request.....                                 | 28        |
| 6.4.9    | Encryption key exchange response .....                               | 29        |
| 6.4.10   | Hash selection request .....                                         | 29        |
| 6.4.11   | Hash selection response .....                                        | 30        |
| 6.4.12   | Path supervision request .....                                       | 30        |
| 6.4.13   | Path supervision response.....                                       | 31        |
| 6.4.14   | Set time command.....                                                | 31        |
| 6.4.15   | Set time response.....                                               | 31        |
| 6.4.16   | Protocol version request .....                                       | 32        |
| 6.4.17   | Protocol version response .....                                      | 32        |
| 6.4.18   | Transparent message.....                                             | 33        |
| 6.4.19   | Transparent response.....                                            | 33        |
| 6.4.20   | DTLS completed request.....                                          | 34        |
| 6.4.21   | DTLS completed response .....                                        | 34        |
| 6.4.22   | RCT IP parameter request.....                                        | 35        |
| 6.4.23   | RCT IP parameter response.....                                       | 35        |
| <b>7</b> | <b>Commissioning and connection setup .....</b>                      | <b>36</b> |
| 7.1      | Commissioning.....                                                   | 36        |
| 7.1.1    | General.....                                                         | 36        |
| 7.1.2    | Procedures .....                                                     | 36        |
| 7.1.3    | Commissioning message sequence .....                                 | 36        |
| 7.1.4    | Commissioning using Shared Secret.....                               | 37        |
| 7.1.5    | Commissioning using X.509 Certificates and DTLS .....                | 38        |
| 7.2      | Connection setup .....                                               | 39        |
|          | <b>Annex A (normative) Result codes .....</b>                        | <b>41</b> |
|          | <b>Annex B (normative) Protocol Identifiers .....</b>                | <b>42</b> |
|          | <b>Annex C (normative) Shared secret .....</b>                       | <b>43</b> |
| C.1      | Formatting of the shared secret.....                                 | 43        |
| C.2      | Checksum for Shared Secret Formatting.....                           | 43        |
| C.3      | Example of Secret Encoding and Formatting .....                      | 43        |
|          | <b>Annex D (informative) Examples of messaging sequences .....</b>   | <b>44</b> |
| D.1      | Commissioning .....                                                  | 44        |
| D.2      | Connection setup.....                                                | 48        |
|          | <b>Annex E (informative) Examples of application protocols .....</b> | <b>51</b> |
| E.1      | SIA.....                                                             | 51        |
| E.2      | Ademco Contact ID.....                                               | 51        |
| E.3      | Scancom Fast Format .....                                            | 52        |
| E.4      | VdS 2465 .....                                                       | 52        |
|          | <b>Annex F (informative) Design principles .....</b>                 | <b>54</b> |
| F.1      | General.....                                                         | 54        |

|                                 |    |
|---------------------------------|----|
| F.2 Information security .....  | 54 |
| F.3 Use of UDP signalling ..... | 54 |
| Bibliography.....               | 55 |

## Tables

|                                                                 |    |
|-----------------------------------------------------------------|----|
| Table 1 — Backwards compatibility .....                         | 9  |
| Table 2 — Backwards compatibility result code .....             | 9  |
| Table 3 — Identifiers .....                                     | 9  |
| Table 4 — Basic unencrypted format of messages .....            | 10 |
| Table 5 — Basic encrypted format of messages .....              | 10 |
| Table 6 — Message ID overview .....                             | 12 |
| Table 7 — Flags .....                                           | 13 |
| Table 8 — Hashing ID's .....                                    | 14 |
| Table 9 — Encryption ID's .....                                 | 15 |
| Table 10 — Reverse commands .....                               | 16 |
| Table 11 — Initial values .....                                 | 17 |
| Table 12 — Poll message SPT $\leftarrow \rightarrow$ RCT .....  | 17 |
| Table 13 — Poll response RCT $\leftarrow \rightarrow$ SPT ..... | 18 |
| Table 14 — Poll response - result code .....                    | 18 |
| Table 15 — Event message format – SPT $\rightarrow$ RCT .....   | 19 |
| Table 16 — Event message format – Fields .....                  | 19 |
| Table 17 — Event field .....                                    | 20 |
| Table 18 — Time event field .....                               | 20 |
| Table 19 — Time message field .....                             | 21 |
| Table 20 — Link field – IP Address .....                        | 21 |
| Table 21 — Link field – Port number .....                       | 21 |
| Table 22 — Link field – URL .....                               | 21 |
| Table 23 — Link field – Filename .....                          | 22 |
| Table 24 — Alarm Text .....                                     | 22 |
| Table 25 — Site Name .....                                      | 22 |
| Table 26 — Building Name .....                                  | 22 |
| Table 27 — Location .....                                       | 23 |
| Table 28 — Room .....                                           | 23 |
| Table 29 — Alarm Trigger .....                                  | 23 |
| Table 30 — Longitude .....                                      | 23 |
| Table 31 — Latitude .....                                       | 24 |
| Table 32 — Altitude .....                                       | 24 |
| Table 33 — Event response message format .....                  | 24 |
| Table 34 — Event response - result code .....                   | 24 |
| Table 35 — Connection handle request message format .....       | 25 |
| Table 36 — Connection handle response message format .....      | 26 |
| Table 37 — Connection handle response - result code .....       | 26 |
| Table 38 — Device ID request message format .....               | 26 |
| Table 39 — Device ID request flags .....                        | 27 |
| Table 40 — Device ID response message format .....              | 27 |
| Table 41 — Encryption selection request message format .....    | 27 |
| Table 42 — 'Master Encryption Selection request' flag .....     | 28 |

|                                                                      |    |
|----------------------------------------------------------------------|----|
| Table 43 — Encryption selection response message format .....        | 28 |
| Table 44 — Encryption selection response - result code .....         | 28 |
| Table 45 — Encryption key exchange request message format .....      | 28 |
| Table 46 — 'Master Key request' flag .....                           | 29 |
| Table 47 — Encryption key exchange response message format .....     | 29 |
| Table 48 — Encryption key - result code .....                        | 29 |
| Table 49 — Hash selection request message format .....               | 30 |
| Table 50 — Hash selection response message format .....              | 30 |
| Table 51 — Path supervision request message format .....             | 30 |
| Table 52 — Path supervision response message format .....            | 31 |
| Table 53 — Path supervision response - result code .....             | 31 |
| Table 54 — Set time command message format .....                     | 31 |
| Table 55 — Set time response message format .....                    | 32 |
| Table 56 — Set time response - result code .....                     | 32 |
| Table 57 — Protocol version request message format .....             | 32 |
| Table 58 — Protocol version response message format .....            | 33 |
| Table 59 — Protocol version response - result code .....             | 33 |
| Table 60 — Transparent message format .....                          | 33 |
| Table 61 — Transparent response format .....                         | 33 |
| Table 62 — Transparent response - result code .....                  | 34 |
| Table 63 — DTLS completed request message format .....               | 34 |
| Table 64 — DTLS completed response message format .....              | 34 |
| Table 65 — DTLS completed response - result code .....               | 34 |
| Table 66 — RCT IP parameter request message format .....             | 35 |
| Table 67 — RCT IP parameter response message format .....            | 35 |
| Table 68 — RCT IP parameter response - result code .....             | 35 |
| Table 69 — Message flow during the commissioning of a new SPT .....  | 36 |
| Table 70 — Message flow during connection setup .....                | 40 |
| Table A.1 — Result codes .....                                       | 41 |
| Table B.1 — Protocol identifiers .....                               | 42 |
| Table D1 — Example of the commissioning messaging sequence .....     | 45 |
| Table D.2 — Example of the connection setup messaging sequence ..... | 48 |
| Table E.1 — VdS2465 message example .....                            | 53 |

## European foreword

This document (CLC/TS 50136-9:2017) has been prepared by CLC/TC 79 “*Alarm systems*”.

This document supersedes CLC/TS 50136-9:2013.

This technical specification specifies a common IP transport protocol for alarm transmission. The published version (2013, first version) required solving both technical and security issues identified during the first actual implementations of the protocol. The working group was working closely with the early adopters of the protocol and has a very clear and complete list of issues and solutions. This revision supersedes the previous version.

EN 50136 will consist of the following parts, under the general title “Alarm systems - Alarm transmission systems and equipment”:

- Part 1            General requirements for alarm transmission systems
- Part 2            General requirements for Supervised Premises Transceiver (SPT)
- Part 3            Requirements for Receiving Centre Transceiver (RCT)
- Part 4            Annunciation equipment used in alarm receiving centres
- Part 5            (Free) **ITeh STANDARD PREVIEW**
- Part 6            (Free) **(standards.iteh.ai)**
- Part 7            Application guidelines
- Part 8            (Free) <https://standards.iteh.ai/catalog/standards/sist/b97f1291-3021-4bb9-b46f-3a69d17a558b/sist-ts-clc-ts-50136-9-2017>
- Part 9            Requirements for a common protocol for alarm transmission using the Internet Protocol (IP)



## 1 Scope

This Technical Specification specifies a protocol for point-to-point transmission of alarms and faults, as well as communications monitoring, between a Supervised Premises Transceiver and a Receiving Centre Transceiver using the Internet Protocol (IP).

The protocol is intended for use over any network that supports the transmission of IP data. These include Ethernet, xDSL, GPRS, WiFi, UMTS and WIMAX.

The system performance characteristics for alarm transmission are specified in EN 50136-1.

The performance characteristics of the supervised premises equipment should comply with the requirements of its associated alarm system standard and applies for transmission of all types of alarms including, but not limited to, fire, intrusion, access control and social alarms.

Compliance with this Technical Specification is voluntary.

## 2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

EN 50136-1:2012, *Alarm systems - Alarm transmission systems and equipment - Part 1: General requirements for alarm transmission systems*

## 3 Terms, definitions and abbreviations

### 3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in EN 50136-1:2012 apply.

### 3.2 Abbreviations

For the purposes of this document, the following abbreviations apply.

|      |                                                |
|------|------------------------------------------------|
| AES  | Advanced Encryption Standard                   |
| ARC  | Alarm Receiving Centre                         |
| ATP  | Alarm Transmission Path                        |
| ATS  | Alarm Transmission System                      |
| CA   | X.509 Certificate Authority                    |
| CBC  | Cipher Block Chaining                          |
| CRC  | Cyclic redundancy check                        |
| DNS  | Domain Name System                             |
| DTLS | Datagram Transport Layer Security              |
| HL   | Header Length                                  |
| IP   | Internet Protocol                              |
| IV   | Initialization Vector                          |
| MAC  | Media Access Control                           |
| MTU  | Maximum Transmission Unit                      |
| NAT  | Network Address Translation                    |
| NIST | National Institute of Standards and Technology |
| NTP  | Network Time Protocol                          |

## CLC/TS 50136-9:2017 (E)

|       |                                      |
|-------|--------------------------------------|
| NVM   | Non-Volatile Memory                  |
| P-MTU | Path Maximum Transmission Unit       |
| RCT   | Receiver Centre Transceiver          |
| RX    | Receive                              |
| SCTP  | Stream Control Transmission Protocol |
| SNTP  | Simple Network Time Protocol         |
| SPT   | Supervised Premises Transceiver      |
| TFTP  | Trivial File Transfer Protocol       |
| TX    | Transmit                             |
| UDP   | User Datagram Protocol               |
| URI   | Uniform Resource Identifier          |
| URL   | Uniform Resource Locator             |
| UTC   | Coordinated Universal Time           |
| WS    | Window Size                          |

## 4 Objective

The object of this Technical Specification is to specify the protocol details (transport and application layers) for alarm transmission systems using Internet Protocol (IP), to ensure interoperability between SPTs and RCTs supplied by different manufacturers. Mechanisms to commission SPT and RCT and build mutual trust between the communicating parties are also described.

As compliance with this Technical Specification is voluntary, any other alarm transmission protocol or equipment not covered by this Technical Specification may be used, provided that the requirements of EN 50136-1 are met.

The protocol and its elements concern one ATP only.

This protocol is designed to run on top of UDP and is designed to support both IPv4 and IPv6.

NOTE For further discussion of IP and UDP in alarm transmission please see F.3.

## 5 Messaging

### 5.1 General

This clause defines the messaging layer, on top of which the alarm event data are transmitted using the existing reporting formats like for example Sia and Contact ID. Clause 7 defines the initial commissioning of an SPT, as well as how SPTs connect to the RCT.

The functionality of the alarm messaging and polling protocol includes:

- exchanging master and session parameters;
- (alarm) event reporting (including linking to out-of-band additional data related to events, like audio/video);
- line monitoring;
- transparent message transmission, e.g. vendor specific messages that, for example, can be used for remote commands from RCT to SPT.

It fulfils the following requirements:

- encryption, fulfilling requirements for most demanding category of EN 50136-1;
- authentication, fulfilling requirements for most demanding category of EN 50136-1;
- SPT: allows a broad range of hardware (limited demands on memory footprint as well as CPU power);
- RCT: allows support for at least 10 000 SPTs in compliance with any category in EN 50136-1, using modern general purpose server hardware;
- allow Dynamic IP addresses of the SPTs;
- allow one or more SPTs to be placed behind a NAT firewall.

## 5.2 Message format overview

### 5.2.1 General

This subclause describes the basic outline of all messages.

Each message shall be explicitly acknowledged, including line supervision messages.

Backwards compatibility is achieved by the implementation of a response with the same message id as the unknown message, however with bit 7 set, with the result code:

“RESP\_CMD\_NOT\_SUPPORTED”.

Table 1 — Backwards compatibility

| Byte index | Bytes | Description                           |
|------------|-------|---------------------------------------|
| 0          | HL    | Header, Message ID and Message Length |
| HL         | 1     | Result code                           |
|            |       | Padding                               |
|            |       | Hash                                  |

Table 2 — Backwards compatibility result code

| Description            | Purpose Description                           |
|------------------------|-----------------------------------------------|
| RESP_CMD_NOT_SUPPORTED | If the message ID is not supported by the RCT |

Multi-byte values will be transmitted using network byte order (big-endian).

### 5.2.2 Identifiers

The following identifiers exist:

Table 3 — Identifiers

| Description       | Purpose                                      | Present in   | Encrypted | See   |
|-------------------|----------------------------------------------|--------------|-----------|-------|
| Connection Handle | Look up the current symmetric encryption key | All messages | No        | 5.2.4 |
| Device ID         | Uniquely identify the hardware               |              | N / A     | 5.2.5 |

## CLC/TS 50136-9:2017 (E)

The Connection Handle is unencrypted. It is a unique number, initialized during the setup of the connection. Its sole purpose is to be able to look up the encryption key. It is valid for the communication session only.

The Device ID uniquely identifies the hardware once the connection has been established.

NOTE Device ID is not equivalent to any account code or similar ID specified by application protocol.

The Device ID shall be stored in non-volatile memory within the SPT.

The IP address is not used for identification purposes, in order to allow for the use of dynamic or translated IP addresses.

### 5.2.3 Message format

The basic unencrypted format of all messages is as follows. Message in this format is never transmitted. It is described here only to clarify the hash value calculation.

**Table 4 — Basic unencrypted format of messages**

| Byte Index | Bytes | Description             | See      | Group   |
|------------|-------|-------------------------|----------|---------|
| 0          | 4     | Connection handle       | 5.2.4    | Header  |
| 4          | 2     | Tx Sequence number      | 5.2.8    |         |
| 6          | 2     | Rx Sequence number      | 5.2.8    |         |
| 8          | 2     | Flags                   | 5.2.9    |         |
| 10         | 1     | Protocol version number | 5.7      |         |
| 11         | 1     | Message ID              | 5.2.6    |         |
| 12         | 2     | Message Length          | 5.2.7    | Message |
| 14         | n     | Message data            | Clause 6 |         |
| 14 + n     | n     | Padding                 | 6.3      |         |

The basic encrypted, transmitted format of all messages is as follows. Note that the Device ID field is not included in the encrypted message and its value is not used to compute the message hash value i.e. the hash is calculated from the unencrypted version of the message described above.

**Table 5 — Basic encrypted format of messages**

| Byte Index | Bytes    | Description                             | See      | Encrypted | Group   |
|------------|----------|-----------------------------------------|----------|-----------|---------|
| 0          | 4        | Connection Handle                       | 5.2.4    | No        | Header  |
| 4          | 2        | Tx Sequence number                      | 5.2.8    | Yes       |         |
| 6          | 2        | Rx Sequence number                      | 5.2.8    | Yes       |         |
| 8          | 2        | Flags                                   | 5.2.9    | Yes       |         |
| 10         | 1        | Protocol version number                 | 5.7      | Yes       |         |
| 11         | 1        | Message ID                              | 5.2.6    | Yes       |         |
| 12         | 2        | Message Length                          | 5.2.7    | Yes       | Message |
| 14         | n        | Message data                            | Clause 6 | Yes       |         |
| 14 + n     |          | Padding                                 | 5.3.1    | Yes       |         |
|            | 32<br>32 | Hash – SHA-256, or<br>Hash – RIPEMD-256 | 5.4      | Yes       | Tail    |

The Connection Handle is unencrypted; the remainder of the message (from Tx Sequence Number up to and including Hash) is encrypted using the encryption method as negotiated during the commissioning stage.

Message ID's are defined in pairs: each message has its matching response. For responses the first byte of the Message Data always holds a 'Result code' as defined in Annex A.

All fields are described in detail in the following subclauses.

#### 5.2.4 Connection handle

The Connection Handle is assigned (uniquely for the RCT to which a SPT reports) using the commissioning protocol. The RCT creates a unique Connection Handle and links this to the Device ID of the SPT in its internal database. This translation results in a compact, fixed length Connection Handle.

The purpose of the Connection Handle is to be able to determine the encryption key to be used to decrypt the received message, independent of the IP address of the message.

The Connection Handle is a 32-bit binary value.

For the purpose of commissioning a SPT, the RCT will present a one-time-connection-handle to the installer/operator. It will be shared/represented as a hexadecimal number (consisting of eight characters).

The final Connection Handle as in use after the commissioning phase is not a (by the installer/operator) configurable parameter, nor made visible on user interfaces. It is used internally by the SPT/RCT equipment only.

#### 5.2.5 Device ID

##### 5.2.5.1 General

The Device ID uniquely identifies the SPT.

Within the message header, the Device ID itself is never transmitted.

Device ID is 16 bytes long.

##### 5.2.5.2 SPT Device ID

The Device ID of the SPT is an ID that is random to the SPT, but fixed and read-only over the lifetime of the SPT, i.e. a hardware serial number. It is unique within the SPT database in the RCT.

The Device ID is created during manufacturing time of the device; in messaging, it is never transmitted itself in cleartext, but is needed to be known in cleartext for the ARC to configure the RCT accordingly.

Thus, it is only transmitted during initial commissioning phase to the RCT.

Uniqueness is ensured by the following principles:

- Each SPT manufacturer shall use his 24 bits "Organizationally Unique Identifier" as assigned to him by the IEEE for MAC-address generation.
- Each SPT manufacturer not having such a code shall attend for such a code from IEEE.
- If an interface in the SPT makes use of a MAC address, the next 24 bits in the device ID shall be the same as the rest of MAC address specified by the manufacturer. If such interface does not exist, the manufacturer shall use another numbering scheme documented by the manufacturer.
- The manufacturer shall use non-consecutive, randomly distributed numbers for the rest of the device ID field and guarantee uniqueness for all his delivered SPT devices.

### 5.2.6 Message ID

The Message ID's as used are listed in the following table:

**Table 6 — Message ID overview**

| Message name          | Description                      | Direction<br>SPT ← →<br>RCT | Version | Message ID |
|-----------------------|----------------------------------|-----------------------------|---------|------------|
| POLL_MSG              | Poll message                     | ← →                         | 1       | 0x11       |
| EVENT_MSG             | Event message                    | →                           | 1       | 0x30       |
| CONN_HANDLE_REQ       | Connection handle request        | →                           | 1       | 0x40       |
| DEVICE_ID_REQ         | Device ID request                | →                           | 1       | 0x41       |
| ENCRYPT_SELECT_REQ    | Encryption selection request     | →                           | 1       | 0x42       |
| ENCRYPT_KEY_REQ       | Encryption key exchange          | ← →                         | 1       | 0x43       |
| HASH_SELECT_REQ       | Hash selection request           | →                           | 1       | 0x44       |
| PATH_SUPERVISION_REQ  | Path supervision request         | ← →                         | 1       | 0x45       |
| SET_TIME_CMD          | Set time command                 | ←                           | 1       | 0x47       |
| VERSION_REQ           | Protocol version request         | →                           | 1       | 0x48       |
| DTLS_COMPLETE_REQ     | DTLS completed request           | →                           | 1       | 0x62       |
| RCT_IP_REQ            | RCT IP parameter request         | →                           | 1       | 0x63       |
| TRANSPARENT_MSG       | Transparent message              | ← →                         | 1       | 0x70       |
| POLL_RESP             | Poll response                    | ← →                         | 1       | 0x91       |
| EVENT_RESP            | Event response                   | ←                           | 1       | 0xB0       |
| CONN_HANDLE_RESP      | Connection handle response       | ←                           | 1       | 0xC0       |
| DEVICE_ID_RESP        | Device ID response               | ←                           | 1       | 0xC1       |
| ENCRYPT_SELECT_RESP   | Encryption selection response    | ←                           | 1       | 0xC2       |
| ENCRYPT_KEY_RESP      | Encryption key exchange response | ← →                         | 1       | 0xC3       |
| HASH_SELECT_RESP      | Hash selection response          | ←                           | 1       | 0xC4       |
| PATH_SUPERVISION_RESP | Path supervision response        | ← →                         | 1       | 0xC5       |
| SET_TIME_RESP         | Set time response                | →                           | 1       | 0xC7       |
| VERSION_RESP          | Protocol version response        | ←                           | 1       | 0xC8       |
| DTLS_COMPLETE_RESP    | DTLS completed response          | ←                           | 1       | 0xE2       |
| RCT_IP_RESP           | RCT IP parameter response        | ←                           | 1       | 0xE3       |
| TRANSPARENT_RESP      | Transparent response             | ← →                         | 1       | 0xF0       |

The Message ID of any Response is the same as the Message ID of the corresponding Command, but with bit 7 set.

### 5.2.7 Message Length

This is the length of the Message Data (excluding Message ID and Message Length). This field is used:

- in variable length messages (see for example 6.3.1 and 6.4.19) to check for the end of data;
- to be able to determine the start of an embedded reverse command (see 5.8).

Possible padding is never considered when calculating the value of message length field.

### 5.2.8 Sequence numbers

The sequence number is used to determine if a message is missing or duplicated. Both ends have a transmit sequence number and a receive sequence number.

These two counters exist at both ends (e.g. we are speaking about 4 counters in total), whereas the RX\_Sequence counters are used to realize a “state-full machine” implementation.

These counters are used to fulfil three simultaneous functions:

- a) Initially, both the SPT and RCT choose their TX\_seqs to be a random number, then they use it as a datagram counter, incrementing them for each sent datagram by one. The RX\_seqs are the expected next TX\_seqs from the other communication end-point. That is: If one did see “42” as the last TX\_seq coming in from the communication partner, oneself would send out “43” as next RX\_seq. As the other end does this in the same style, the TX\_seq and RX\_seq function as a mutual sequence control mechanism.
- b) Second, they can simultaneously function as a resend-mechanism: If one detected that one missed a datagram (because for example, the incoming TX\_seq is “44”, but one expected TX\_seq = 43) or the one got is corrupt (by checking the hash), one just resends the own old previously sent last datagram and the other side will see by the old TX\_seq that one wants to get a re-transmission.
- c) Being chosen randomly and being part of the encrypted data block, they rule out replay attacks.

For each connection, every message shall be acknowledged before the next new (not retransmission) message may be transmitted.

### 5.2.9 Flags

The following flags are defined:

**Table 7 — Flags**

| Byte | Bit   | Definition                                                                                                                |
|------|-------|---------------------------------------------------------------------------------------------------------------------------|
| 0    | 0     | Reverse command included in response:<br>– value 0 = no reverse command included,<br>– value 1 = reverse command included |
|      | 1...7 | Reserved                                                                                                                  |
| 1    | 0...7 | Reserved                                                                                                                  |

## 5.3 Padding and message length

### 5.3.1 General

Padding is required for the following two reasons:

- create a message length which is a multiple of the block length of the encryption algorithm as used;