ERTMS/ETCS

ERTMS End-to-End Security

REF : SUBSET-146
ISSUE : 4.0.0
DATE : 05-07-2023

Company	Technical Approval	Management approval
ALSTOM		
AZD		
CAF		
HITACHI RAIL STS		
MERMEC		
SIEMENS		
THALES		

1. MODIFICATION HISTORY

Issue Number Date	Section Number	Modification / Description	Author
0.1.0 (03-10-2022)		All EECT review completed	PL
3.9.2 (23-02-2023)		Only new version number (No change in document content)	PL
3.9.3 (31-05-2023)	Sections 3.1.1.4, 3.1.1.10, 3.1.1.11, 5.6.3.2.2, Tables 5, 11, 16, 17, Annex A2, A3, A4	Updated following EECT meeting on 14th March 23 and ERA comments (Outcome of B4R1 3rd consolidation phase)	PL
3.9.4 (02-07-2023)	Document name	ERA comment	PL
4.0.0 (05-07-2023)	-	Baseline 4 1st release version	GP

2. TABLE OF CONTENTS

3. INTRODUCTION

3.1 Scope and Purpose

3.1.1.1 ERTMS applications exchange information, e.g., between on-board and trackside entities, over open transmission systems.
Note: The exhaustive application cases, i.e., KM, ATO and ATP will be covered in ANNEX A.

3.1.1.2 Data transmission over open transmission systems are inherently vulnerable as unauthorised access cannot be excluded. Therefore, it is important to guarantee the information security for the transmitted data.

3.1.1.3 The typical IP communication stack used for ERTMS data applications: ETCS, ATO and KM is shown in Figure 1 below.

3.1.1.4 The safety layer shown in Figure 1 is only applicable for safety related communication, i.e., for the ATP application.

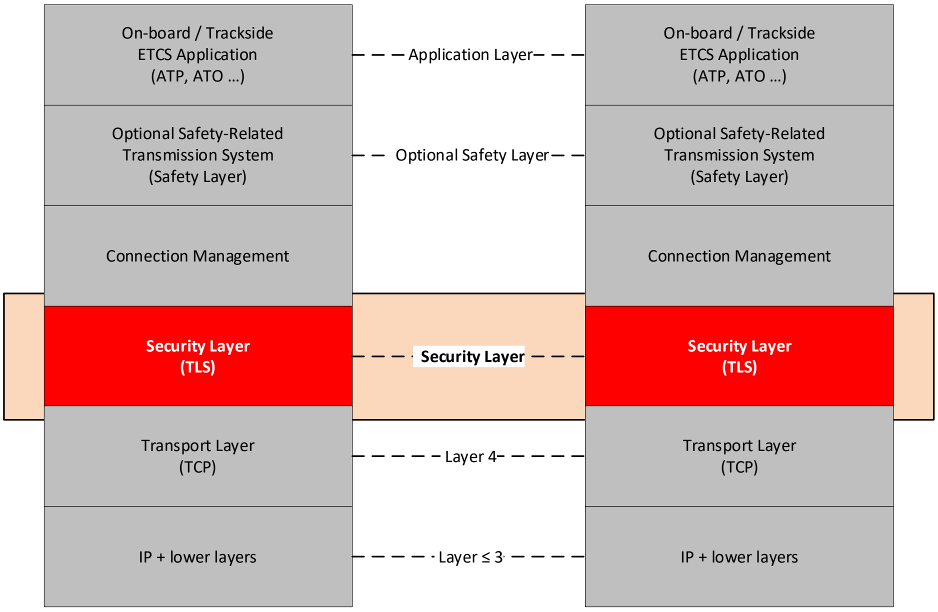

Figure 1. ERTMS Application Communication

3.1.1.5 This Subset specifies interoperable standards and requirements for protecting the transmitted information, i.e., the security layer shown in Figure 1.

3.1.1.6 The separation of safety and security layer is an enabler to keep security of the communication updated, without the need to re-approve any safety rating of the application.

3.1.1.7 An overview of the used cyber security functions is presented in chapter 4.

3.1.1.8 Requirements are specified in chapter **Error! Reference source not found.**.

3.1.1.9 The requirements are targeting interoperable end-to-end security by use of TLS.
Note: Secure implementation relies on adequate security design, which is not subject of this specification. However, recommendations are given on dedicated topics.

3.1.1.10 ANNEX A show which requirements that are applicable for respective ERTMS data applications: ETCS, ATO and KM.

3.1.1.11 This specification is applicable for communication over packet switched bearer(s). Note: subset-146 is not applicable for ETCS over GPRS. For this use case, the Euroradio protocol stack defined in SS-037-1 remains applicable.

3.2 References

50701.C.4	Railway applications — Cybersecurity PD CLC/TS 50701:2021 Annex C-4	2021
ANSSI	Security Recommendations for TLS SDE-NT-35-EN/ANSSI/SDE/NP	January 2017
OpenSSL	https://www.openssl.org/	January 2022
RFC-3647	Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework	November 2003
RFC-3779	X.509 Extensions for IP Addresses and AS Identifiers	June 2004
RFC-4055	Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile	June 2005
RFC-4158	Internet X.509 Public Key Infrastructure: Certification Path Building	September 2005
RFC-4210	Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)	September 2005
RFC-4211	Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)	September 2005
RFC-4279	Pre-Shared Key Cypher Suites for Transport Layer Security (TLS)	December 2005
RFC-4086	Randomness Requirements for Security	June 2005
RFC-4492	Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)	May 2006
RFC-5280	Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile	May 2008
RFC-5424	The Syslog Protocol	March 2009
RFC-5425	Transport Layer Security (TLS) Transport Mapping for Syslog	March 2009
RFC-5480	Elliptic Curve Cryptography Subject Public Key Information	March 2009
RFC-5487	Pre-Shared Key Cipher Suites for TLS with SHA-256/384 and AES Galois Counter Mode	March 2009
RFC-5639	Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation	March 2010
RFC-5758	Internet X.509 Public Key Infrastructure: Additional Algorithms and Identifiers for DSA and ECDSA	January 2010
RFC-5905	Network Time Protocol Version 4	June 2010
RFC-6712	Internet X.509 Public Key Infrastructure – HTTP Transfer for the Certificate Management Protocol (CMP)	September 2012

RFC-6818	Updates to the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile	January 2013
RFC-6960	X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP	June 2013
RFC-8446	The Transport Layer Security (TLS) Protocol Version 1.3	August 2018
RFC-8813	Clarifications for Elliptic Curve Cryptography Subject Public Key Information	August 2020
RFC-9150	TLS 1.3 Authentication and Integrity-Only Cipher Suites	January 2022
Subset-023	ERTMS/ETCS; Glossary of Terms and Abbreviations	
Subset-037-1	EuroRadio FIS CS/PS Communication Functional Module	
Subset-037-3	EuroRadio FIS FRMCS Communication Functional Module	
Subset-137	On-line Key Management FFFIS	
Subset-148	ATO-OB / ATO-TS Interface Specification Transport and Security Layer	
x509errors	https://x509errors.org/	January 2022
X.500	ITU-T Recommendation: Information technology - Open Systems Interconnection - The Directory: Overview of concepts, models and services	October 2012
X.501	ITU-T Recommendation: Information technology - Open Systems Interconnection - The Directory: Models	October 2012
X.520	ITU-T Recommendation: Information technology - Open Systems Interconnection - The Directory: Selected attribute types	October 2012
X.690	ITU-T Recommendation: Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)	July 2002

3.3 Acronyms and abbreviations

3.3.1.1 For general ERTMS/ETCS acronyms and abbreviations see [Subset-023]. Additional acronyms and abbreviations relevant for this document are specified here.

Abbreviation	Definition
CA	Certificate Authority
CEF	Common Event Format
CERT	Cybersecurity Emergency Response Team
CMP	Certificate Management Protocol
CRL	Certificate Revocation List
DANE	DNS-based Authentication of Named Entities
DN	Distinguished Name

DNS	Domain Name Service
ECC	Elliptic Curve Cryptography
ECDH	Elliptic Curve Diffie Hellman
EE	End Entity
FRMCS	Future Rail Mobile Communication System
OCSP	On-line Certificate Status Protocol
PKI	Public Key Infrastructure
PSK	Pre-Shared Key
RA	Registration Authority
RTC	Real Time Clock
TLS	Transport Layer Security
TLSA	TLS Authentication record
TSI	Technical Specifications for Interoperability
UTC	Coordinated Universal Time
UTF-16	Unicode Transformation Format 16-bit

3.4 Terms and Definitions

3.4.1.1 For general ERTMS/ETCS terms and definitions see [Subset-023]. Additional terms and definitions relevant for this document are specified here.

Term	Definition
Authentication	The process to verify the identity of communicating peers.
Certificate Authority	The entity responsible for issuing digital certificates to associate public keys with user identities
Confidentiality	Confidentiality, in the context of computer systems, allows only authorised users to access protected data using specific mechanisms to ensure confidentiality and safeguard data from harmful intrusion
Cryptography	The principles, means and methods for transformation of data in order to ensure confidentiality, authenticity, non-repudiation and integrity
End-to-End Security	In the context of ERTMS applications, end-to-end security means protection of the communication between communicating parties. Risks at the communications endpoints themselves are not addressed.
Integrity	Transmitted data cannot be modified by and unauthorised user without detection
Open Transmission System	A transmission system where unauthorised access cannot be excluded, i.e., one or more of the following properties: authenticity, integrity, timeliness and sequence of messages are not protected.
Pseudorandom	A pseudorandom number generator is an algorithm for generating number generator a sequence of numbers whose properties approximate the properties of sequences of random numbers
Registration Authority	The responsible entity in a PKI for accepting requests for digital certificates and authenticating the entity making the request
Security	Condition of system resources being free from unauthorised access and from unauthorised or accidental change, destruction, or loss. Specifically in the context of this subset, the function to provide a secure channel between two communicating peers. The channel is authenticated, integrity protected and optionally private.

Security Infrastructure	The set of hardware, software, people, policies and procedures needed to manage the registration of entities and distribution and storage of digital certificates in a system
System-wide time service	Service that provides synchronic UTC time to onboard and trackside clients.
TLS-PKI user	A user acting as either TLS client or TLS server and using security certificates managed by a public key infrastructure to identify and authenticate the user in question. The private and public keys of the certificate(s) is used to determine the encryption keys for the TLS communication.
TLS-PSK user	A user acting as either TLS client or TLS server and using a pre-shared key to determine the encryption keys for the TLS communication.

4. CYBER SECURITY FUNCTIONS

4.1 General Information

4.1.1.1 This chapter (chapter 4) contains an overview of used cyber security functions.

4.1.1.2 Detailed requirements are specified in chapter 5.

4.2 Public Key Infrastructure

4.2.1.1 Public Key Infrastructure (PKI) is a set of processes, policies, and technology for associating cryptographic keys with the entity to whom those keys were issued. It is a standardized method used for authentication and encryption to confirm the identity of communicating parties as well as validate information being shared.
See also [RFC-3647] and [RFC-4158].

4.2.1.2 Internet protocols intended to provide security for information exchange, for example TLS, may use PKI certificates to authenticate communicating parties with each other as well as support encryption of the communication session.

4.2.1.3 PKI consist of several elements,

a) X.509 Digital Certificates
A type of certificate that includes information about the identity of the owner of the certificate, a digital signature of the certificate authority and keys for encryption/decryption of data.

b) Certification Authority (CA)
A trusted entity that serves authentication infrastructures as well as registering entities that need PKI. It is the organization that issues out digital certificates.

c) Registration Authority (RA)
Is certified by a CA and validates the identity of PKI users requesting information on a certificate.

d) Certificate Revocation List (CRL)
Provides a means for checking the continued validity of the certificates for which the CA has responsibility. The CRL contains digital certificates that are no longer valid.

e) Online Certificate Status Protocol (OCSP) responder
A service used for obtaining the revocation status of an X.509 digital certificate.

4.2.1.4 The following figure depicts the general PKI certificate hierarchy.

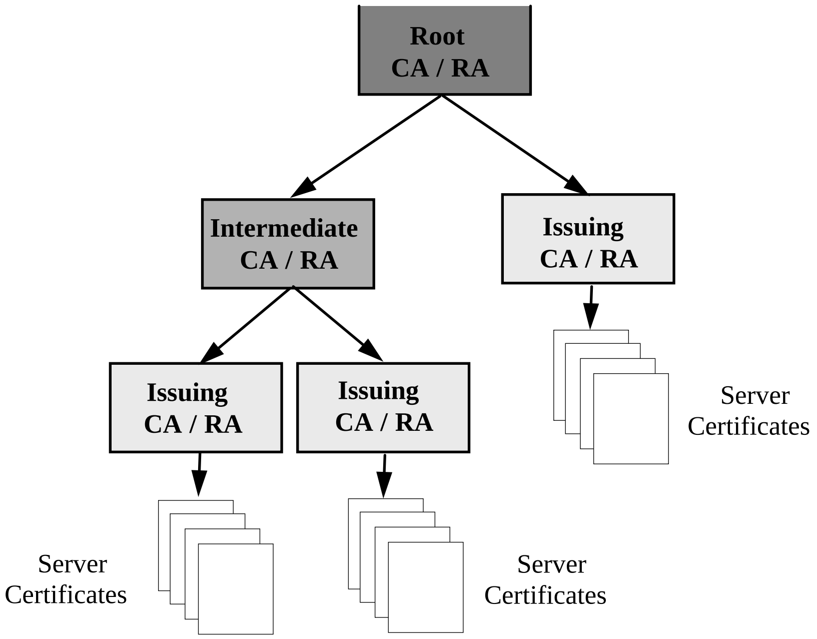

Figure 2 - PKI Hierarchy

4.2.1.5 In the figure above, CA is a Certificate Authority responsible for issuing, renewing, and revoking digital certificates. A digital certificate contains, among others, a public key and information related to the key, its owner, its validity period, and its allowed use (for example encryption and/or authentication).

4.2.1.6 In the simplest scenario, certificates are issued by a Certificate Authority. More complex scenarios see the presence of a Registration Authority (RA). When a new entity wants to obtain a client certificate, it issues a request to the RA which then tries to authenticate the requester. If authenticated, the RA forwards the request to the CA, which issues the digital certificate. The RA can be a part of a CA as well as a separate entity.

4.2.1.7 CAs can be organized in a hierarchical tree structure with CA certificates issued by a higher-level CA. This tree structure has a single root node, called “root CA” and all clients must know the root CA certificate.
Note: A PKI user may be related to more than one root CA.

4.2.1.8 Digital certificates are distributed using the Certificate Management Protocol (CMP).

4.2.1.9 The digital certificate is validated by checking the Certificate Revocation List (CRL) or optionally using the Online Certificate Status Protocol (OCSP).

4.2.1.10 A CRL is downloaded from the URL given by a property of the certificate.

4.2.1.11 The URL to the OCSP server is given by a property of the certificate.

4.2.1.12 The following figure shows interfaces and protocols used between the PKI and PKI users.

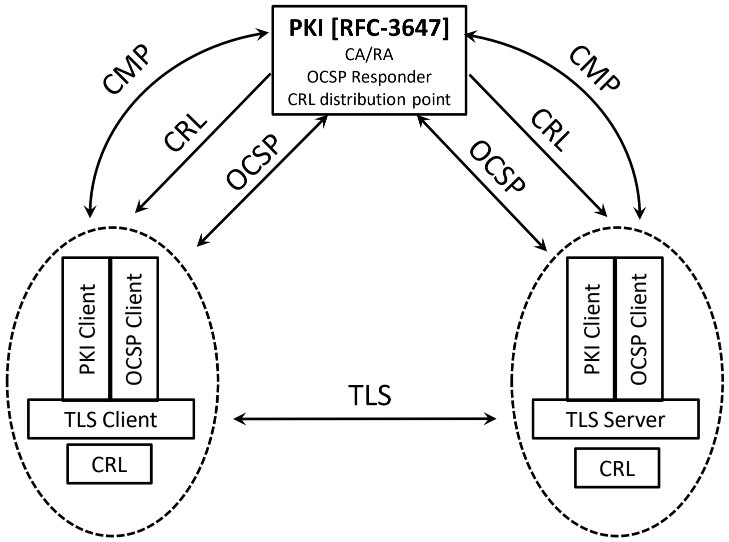

Figure 3. Interfaces between PKI and PKI users

4.2.1.13 The TLS interface allows establishing a TLS connection between two TLS users and securely exchange information over this connection.

4.2.1.14 The certificate delivery interface (CMP in Figure 3) between a PKI client and the PKI allows generating or renewing the certificate of the PKI client using the CMP protocol.

4.2.1.15 The validity of a certificate is checked directly against the CRL produced by the CA. This requires that the PKI client obtains a copy of the CRL and periodically updates the list.

4.2.1.16 The optional interface between an OCSP client (in the PKI user) and an OCSP responder allows an alternative check of the validity of a peer entity certificate using the OCSP protocol.

4.3 Transport Layer Security

4.3.1.1 Transport Layer Security (TLS) is a protocol layer on top of TCP designed to provide confidentiality and data integrity between the communicating parties.

4.3.1.2 TLS protect the communication by the following properties,
- The communication partners are authenticated by using public key cryptography or optionally pre-shared keys.
- The communication is reliable as transmitted data includes a message integrity check using a message authentication code.
- The communication is confidential if the transmitted data is encrypted.

4.3.1.3 At the establishment of the TLS layer the identities of the communicating parties are authenticated.

4.3.1.4 At the establishment of the TLS layer, certificates are checked against the CRL or by using OCSP.

4.3.1.5 The communicating parties negotiate the details of which encryption algorithm and cryptographic key(s) to use at the start of a communication session and before any user data is transmitted.

4.3.1.6 The negotiation is encrypted, and integrity protected, which means that any shared secret cannot be obtained by an attacker and cannot be modified by an attacker without being detected.

4.3.1.7 To ensure confidentiality the data transmitted over a connection can be encrypted using symmetric cryptography.
Note: Asymmetric encryption is used to establish the communication session. After establishment symmetric encryption is used.

4.3.1.8 The key for the symmetric encryption is unique for each session.

4.3.1.9 The connection is integrity protected using a message authentication code to detect loss or alteration of the information during transmission.

4.3.1.10 The TLS protocol using a Public Key Infrastructure is referred to as TLS-PKI throughout the rest of the document.

4.3.1.11 The TLS protocol using pre-shared keys is referred to as TLS-PSK throughout the rest of the document.
Note: TLS-PSK is kept in the subset only for backwards compatibility with Subset-137 v1.0.0. It is not recommended for any new implementation.

4.4 Supporting Services

4.4.1 Random Number Generation

4.4.1.1 The use of digital security certificates and use of secure communication protocols relies upon cryptographically secure random numbers.
Note: Recommendations for implementation regarding random number generation and seeding are given in [RFC-8446] Appendix C.

4.4.2 System-wide time service

4.4.2.1 A system-wide synchronized time is needed for evaluating the validity of certificates and for time stamping events written to security logs.
Note: Requirements are stated in chapter 5.3.

4.4.2.2 System-wide time synchronization is achieved by a central time service that acts as a time source for all components requiring time synchronization.

4.4.2.3 The system-wide time synchronization function typically can synchronize its time with a time source for example from a customer time source or a global navigation satellite system.

4.4.2.4 The time source may be behind a demilitarized zone (DMZ).

5. REQUIREMENTS

5.1 General Information

5.1.1.1 This section (chapter 5) contains requirements on cyber security functions, interoperable interfaces, and protocols.

5.1.1.2 Requirements can be common for several ERTMS applications, or specific per application.

5.1.1.3 The applicability per ERTMS application is specified in ANNEX A.

5.1.1.4 To keep end-2-end communication as secure as possible for ERTMS application it is assumed that implementation and maintenance of the security layer follows best practice regarding vulnerability risk management.

5.2 Random number generation

5.2.1.1 A cryptographically secure random or pseudo-random number generator shall be used to generate:

a) public/private key pair for TLS-PKI
b) pre-shared key when the TLS-PSK option is used
c) session unique keys for TLS

5.2.1.2 Implementation and initialisation of the random number generator shall follow the recommendations given by appendix C.1 in [RFC-8446].

Note: As mentioned in the appendix C.1, additional guidance on the generation of random values is provided by [RFC-4086].

5.3 System-wide time service

5.3.1.1 The protocol for time synchronization shall be NTPv4, see also [RFC-5905].

Note: Subset-146 security required use cases, e.g. verification of certificate and certificate revocation list and logging of security-related events do not need very accurate time synchronization. About +/- 1 seconds is normally sufficient.

5.3.1.2 Time source configuration

5.3.1.2.1 The time source shall be configured to avoid loss of service.
Note: For example, by using redundant time sources, and/or a local RTC module.

5.3.1.2.2 The time source shall manage time jumps.
Note: For example, to keep logged security-related events in chronological order.

5.4 Transport Layer Security (TLS)

5.4.1 TLS overall requirements

5.4.1.1 TLS version 1.2 and 1.3 shall be supported. See [RFC-8446].
Note: Version 1.3 is the preference. Version 1.2 may be used to ensure backwards compatibility to existing implementations.

5.4.1.2 A TLS client shall be able to negotiate which TLS version to use. See [RFC-8446] appendix D.1.

5.4.1.3 A TLS server shall be able to negotiate which TLS version to use. See [RFC-8446] appendix D.2.

5.4.1.4 The zero round-trip time mode (added in TLS 1.3) shall not be used. See [RFC-8446] appendix D.3.

5.4.1.5 End-to-end communicating parties shall be mutually authenticated.

5.4.1.6 The communication shall be integrity protected.
Note: This is the case for the ciphers specified below.

5.4.1.7 The information (application payload) exchanged between communicating parties may be encrypted.
Note: The decision to encrypt the payload is determined from a risk analysis. The TLS server is then configured to select a cipher suite with or without encryption of the application payload. That is, data integrity and authentication shall always be guaranteed, but confidentiality is optional.

5.4.1.8 The TLS communication shall be uncompressed.

5.4.1.9 Resumption of a previous TLS session shall not be allowed by the TLS server.

5.4.1.10 Duplication of an existing TLS session shall not be allowed by the TLS server.

5.4.1.11 Renegotiation of an existing TLS session shall not be allowed.

5.4.1.12 The TLS session shall be released when the TCP connection is disconnected.

5.4.2 TLS requirements for TLS-PKI

5.4.2.1 The certificate of the root Certificate Authority shall be installed in all peer entities using an operational procedure to guarantee its authenticity.
Note: The definition of the operational procedure is out of scope of this document.

5.4.2.2 Authentication between the TLS client and server shall be based on X509 v.3 certificates (see [RFC-5280]).

5.4.2.3 The following cipher suites shall be supported

TLS Version	Cipher Suite	Comment
1.2	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	Preferred for new design Recommendation in [ANSSI]
	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	Used for backwards compatibility
	TLS_ECDHE_ECDSA_WITH_NULL_SHA	Used only when confidentiality is not required
1.3	TLS_AES_256_GCM_SHA384	Preferred
	TLS_CHACHA20_POLY1305_SHA256	Used as backup in case the preferred cipher suite becomes considered insecure
	TLS_SHA384_SHA384	Used only when confidentiality is not required See [RFC-9150]

Table 1. Mandatory Supported Cipher Suites

Note: The TLS server will select one of the supported cipher suites based on configuration of the server, for example in case confidentiality is required a cipher suite with confidentiality will be used.

Note: The concept is to use the most secure configuration shared by both communicating entities.

5.4.2.4 Hello messages

5.4.2.4.1 A TLS client that proposes ECC cipher suites may optionally not include the “Elliptic Curves Extension” and “Supported Point Formats Extension”. In this case, the server shall choose the elliptic curves or point formats.

5.4.2.4.2 The TLS client and TLS server shall support the following elliptic curves,

Curve	Comment
secp256r1	Default curve. Reference [RFC-5480]
brainpoolP256r1	Reference [RFC-5639]. Supported for backwards compatibility reasons.

Table 2. Mandatory Supported Elliptic Curves

5.4.2.4.3 A TLS client that proposes “Multiple Certificate Status Request” extension may provide a zero-length "responder_id_list". In this case, the responders shall be implicitly known by the server, or shall be identified by the certificates used by the server.

5.4.2.4.4 The TLS server shall support “Elliptic Curves Extension” and “Supported Point Formats Extension”.

5.4.2.4.5 The “Supported Point Formats Extension” shall be included in a Server Hello message in response to a Client Hello message containing the “Supported Point Formats Extension” when negotiating an ECC cipher suite.

5.4.2.4.6 A TLS server using TLS 1.2 may support “Multiple Certificate Status Request” extension. In such case the server shall return an extension of type "status_request_v2" with empty "extension_data".

5.4.2.5 Server Certificate

5.4.2.5.1 The “Server Certificate” message shall be used by the TLS server and supported by the TLS client to authenticate the server.

5.4.2.6 Certificate Status

5.4.2.6.1 Certificate status information shall be provided by using certificate revocation lists (CRLs), see [RFC-5280].

5.4.2.6.2 Certificate status information may be provided by using the Online Certificate Status Protocol (OCSP), see [RFC-6960].

5.4.2.6.2.1 The “Certificate Status” message shall be used by the TLS server in order to report the list of OCSP responses for the matching corresponding certificate in the server Certificate in case of use of “Multiple Certificate Status Request” extension.

5.4.2.6.2.2 The periodicity for refreshing the list of OCSP responses is a TLS server configuration parameter. This time period value shall be between 1 hour and 100 hours, with the default value being 10 hours.

5.4.2.6.2.3 In case of successful use of the “Multiple Certificate Status Request” extension, including a freshness check, the TLS client does not need to check the certificate status of the peer entities through OCSP requests.

5.4.2.6.3 If the status of a certificate is revoked or unknown, the connection setup shall be aborted.

5.4.2.7 Server Key Exchange

5.4.2.7.1 For TLS 1.2 the “Server Key Exchange” message shall be used by the TLS server and supported by the TLS client to convey the server’s ephemeral ECDH public key (and the corresponding elliptic curve domain parameters) to the client.

5.4.2.8 Certificate Request

5.4.2.8.1 The “Certificate Request” message shall be used by the TLS server and supported by the TLS client.

5.4.2.8.2 This message shall be extended as specified in § 5.5 of [RFC-4492].

5.4.2.9 Client Certificate

5.4.2.9.1 The “Client Certificate” message shall be used by the TLS client and supported by the TLS server.

5.4.2.9.2 The “Client Certificate” message shall comply with the certificate types listed in the Certificate Request.

5.4.2.10 Client Key Exchange

5.4.2.10.1 For TLS 1.2 the “Client Key Exchange” message shall be used by the TLS client and supported by the TLS server.

5.4.2.10.2 This message shall be extended as specified in § 5.7 of [RFC-4492].

5.4.2.11 Certificate Verify

5.4.2.11.1 The “Certificate Verify” message shall be used by the TLS client and supported by the TLS server.

5.4.2.12 Change Cipher Spec

5.4.2.12.1 For TLS 1.2 the “Change Cipher Spec” message shall be supported and used by both the TLS client and the TLS server.

5.4.3 TLS requirements for TLS-PSK

5.4.3.1 The distribution and installation of secret pre-shared keys for TLS-PSK shall be supported by an operational procedure to guarantee secrecy and authenticity.
Note: Definition of the operational procedure is out of scope of this document.

5.4.3.2 A unique pre-shared key shall be generated by the TLS users. This pre-shared key shall be used to authenticate both peers.

5.4.3.3 Installation of a pre-shared key in a TLS user overwrites any previously stored pre-shared key in the TLS user.

5.4.3.4 The size of the pre-shared key shall be at least 256 bits.

5.4.3.5 The TLS clients and servers shall support at least the following cipher suite: TLS_DHE_PSK_WITH_AES_256_GCM_SHA384 (see [RFC-4279], [RFC-5487]).

5.4.3.6 The TLS client shall provide a “PSK identity” in the ClientKeyExchange message (see [RFC-4279], §2).

5.4.3.7 The TLS server shall provide a PSK identity hint in the ServerKeyExchange message (see [RFC-4279], §2).

5.4.3.8 The expanded ETCS ID of the sender of the TLS message shall be used as the PSK identity and PSK identity hint.

5.5 Certificate delivery interface

5.5.1 General

5.5.1.1 A TLS-PKI user shall establish TCP connections with the PKI to manage digital certificates.

5.5.1.2 The FQDN of RA and CA shall be configured in the TLS-PKI user.

5.5.2 Client certificate delivery functions

5.5.2.1 The following table provides the function allocation for the certificate delivery interface:

Function	Message flow direction			Purpose
Certificate Request	PKI client	→	PKI server	Request a certificate
Certificate Response	PKI server	→	PKI client	Deliver a certificate
Certificate Confirmation	PKI client	→	PKI server	Confirm the reception of a certificate
Confirmation Acknowledgement	PKI server	→	PKI client	Acknowledge the Certificate Confirmation message

Table 3. Functions allocation for the certificate delivery interface

5.5.2.2 The following kinds of certificate generation shall be supported.

5.5.2.2.1 Generation of first certificate with a public key.
Note: First request without any valid certificate.

5.5.2.2.2 Generation of new certificate with a new public key.
Note: Renewal of certificate and public key (rekey)

5.5.2.3 Certificate Request

5.5.2.3.1 The PKI client shall request the PKI server for delivery of a first certificate.

5.5.2.3.2 The PKI client shall request a certificate rekey a configurable amount of time before the expiration of the current certificate.

5.5.2.3.3 It is recommended that a certificate should be valid for 6 months and being renewed 1 month before expirations.

5.5.2.3.4 An alarm shall be generated if a renewal of a certificate is not successful.

5.5.2.3.5 The PKI client shall generate a public/private key pair at first certificate request and at every certificate rekey request.

5.5.2.3.6 The PKI client shall keep its private key secret.

5.5.2.3.7 The public key length for a PKI client shall be for 3072 bits for RSA cryptography. For elliptic curve cryptography, the key length is defined by the used curve.

5.5.2.3.8 The public key length for a Certificate Authority shall be 3072 bits for RSA cryptography. For elliptic curve cryptography, the key length is defined by the used curve.

5.5.2.3.9 Each PKI client shall have its own unique Distinguished Name (DN).

5.5.2.3.10 For the first certificate request, the PKI client shall authenticate itself by using shared secret information (a ‘passphrase’) to create the “protection” field contained in the “Certificate Request” message or by using a valid certificate (e.g., a manufacturer device certificate).

5.5.2.3.11 The characters used for the passphrase shall be encoded using UTF-16 with a minimum length of 16 characters.

5.5.2.3.12 The ‘passphrase’ shall not be part of the initial configuration but shall be provided when needed by a specific process independent from this interface.

5.5.2.3.13 For a certificate rekey request, the PKI client has a valid certificate and shall use this valid certificate to authenticate itself when requesting a new certificate. The PKI client shall create the “protection” field contained in the “Certificate Request” message by using the private key associated to its valid certificate.

5.5.2.4 Certificate Response

5.5.2.4.1 If the PKI server considers a certificate request from a PKI client as valid, the PKI server shall be able to sign and deliver a new certificate to this PKI client.

5.5.2.4.2 If a certificate request is valid, the “Certificate Response” message shall include:

a) a signed certificate corresponding to the template certificate contained in the certificate request.

b) the certificate hierarchy for this certificate, except the root certificate which must be delivered to the PKI client in a secure way (with an organisational process).

5.5.2.4.3 If the certificate request is not valid, the PKI server shall send a negative certificate response.

5.5.2.5 Certificate Confirmation

5.5.2.5.1 A “Certification Confirmation” message shall be sent by the PKI client to the PKI server at reception of the “Certificate Response” message.

5.5.2.6 Confirmation Acknowledgment

5.5.2.6.1 The PKI server shall acknowledge the reception of the confirmation from the PKI client.

5.5.2.7 Unless revoked by the Certificate Authority a certificate is valid to the end of the validity period.

5.5.3 Interface specification

5.5.3.1 General requirements

5.5.3.1.1 The “Certificate Request”, “Certificate Response”, “Certification Confirmation” and “Confirmation Acknowledgement” messages may be exchanged in the same session.

5.5.3.1.2 The HTTP protocol shall be used for transfer of CMP messages, see [RFC-6712].

5.5.3.1.3 In case the process is not completed due to TCP disconnection it shall be considered as failed and the PKI server shall revoke the newly generated certificate.

5.5.3.1.4 The certificate delivered by the PKI server shall conform to X509 v.3 (see [RFC-5280] and [RFC-6818]).

5.5.3.1.5 The “Certificate Request”, “Certificate Response”, “Certification Confirmation” and “Confirmation Acknowledgement” messages shall comply with the CMP protocol (see [RFC-4210] and [RFC-4211]).

5.5.3.1.6 Note: For each optional field in the CMP messages (see Table 4 to Table 18), it is stated if this optional field:

- shall be used
- shall not be used
- may be used

5.5.3.1.7 The used fields shall be present in the message.

5.5.3.1.8 The fields that are not used shall not be present in the message.

5.5.3.2 CMP message specification: PKI message common fields

5.5.3.2.1 The “Certificate Request”, “Certificate Response”, “Certification Confirmation” and “Confirmation Acknowledgement” messages shall comply with Table 4 and Table 5.

PKIMessage (see RFC-4210**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
Header	PKIHeader	Mandatory	Mandatory	See PKIHeader defined in Table 5
Body	CertReqMessage, CertRepMessage, CertConfirmContent, PKIConfirmContent	Mandatory	Mandatory	Shall be one of the following: 1. CertReqMessages (defined in Table 6) for a Certificate Request Message 2. CertRepMessage (defined in Table 12) for a Certificate Response Message 3. CertConfirmContent (defined in Table 18) for a Certificate Confirmation Message 4. PKIConfirmContent (see section 6.3.2.6) for a Confirmation Acknowledgement Message
Protection	BIT STRING	Optional	Mandatory	Signature value with protectionAlg or Shared Secret Information
extraCerts 1..MAX	Certificate	Optional	Mandatory	See Table 15

Table 4. PKIMessage

PKIHeader (See RFC-4210**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
pvno	INTEGER	Mandatory	Mandatory	Must be set to “2”
sender	GeneralName	Mandatory	Mandatory	Distinguished Name shall be used For KM use case: ETCS-ID-EXP
recipient	GeneralName	Mandatory	Mandatory	Distinguished Name shall be used For KM use case: ETCS-ID-EXP

messageTime	GeneralizedTime	Optional	Mandatory	UTC time shall be used
protectionAlg	AlgorithmIdentifier	Optional	Mandatory	For a first certificate request, algorithmIdentifier shall be PasswordBasedMAC, ecdsa-with-SHA256 or ecdsa-with-SHA512 for messages emitted by PKI client As default the algorithmIdentifier for messages emitted by PKI server and rekey messages emitted by PKI client shall be ecdsa-with-SHA256 or ecdsa-with-SHA512 ([RFC-5758]). For backward compatibility reasons (e.g., the KM use case), for messages emitted by PKI server and certificate rekey messages emitted by PKI client the algorithmIdentifier sha384WithRSAEncryption shall be supported ([RFC-4055]).
senderKID	KeyIdentifier	Optional	Mandatory for PKI client	Shall be used for the messages emitted by the PKI client for rekey purpose. Can be used for other purposes
recipKID	KeyIdentifier	Optional	May be used	
transactionID	OCTET STRING	Optional	Mandatory	Set by Client, returned by Server
senderNonce	OCTET STRING	Optional	Mandatory	Sender generated 128bit (pseudo-) random data
recipNonce	OCTET STRING	Optional	Mandatory	Copied from senderNonce of previous message in the transaction
freeText		Optional	Shall not be used	
generalInfo 1..MAX		Optional	Shall not be used	

Table 5. PKIHeader

5.5.3.3 CMP message specification: “Certificate Request” message

5.5.3.3.1 For the “Certificate Request” message, the body of the PKI message shall comply with Table 6, Table 7, Table 8, Table 9, Table 10 and Table 11.

CertReqMessage (see RFC-4211**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
certReqMsg 1..MAX	CertReqMsg	Mandatory	Mandatory	See CertReqMsg defined in Table 6

Table 6. CertReqMessage

CertReqMsg (see RFC-4211**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
certReq	CertRequest	Mandatory	Mandatory	See CertRequest defined in Table 7
popo	ProofOfPossesion	Optional	Mandatory	Shall be a signature of type POPOSigningKey (see Table 10)
regInfo 1..MAX		Optional	Shall not be used	

Table 7. CertReqMsg

CertRequest (see RFC-4211**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
certReqId	Integer	Mandatory	Mandatory	Set by requester
certTemplate	CertTemplate	Mandatory	Mandatory	See CertTemplate defined in Table 8
controls		Optional	Shall not be used	

Table 8. CertRequest

CertTemplate (see RFC-4211**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
version	INTEGER	Optional	Mandatory	Must be set to “2”
serialNumber		Must be omitted	Shall not be used	
signingAlg		Must be omitted	Shall not be used	
issuer		Optional	Shall not be used	
validity		Optional	Shall not be used	
subject	Name	Optional	Mandatory	The ‘Common Name’ defined in table Table 19 shall be used.
publicKey	SubjectPublicKey Info	Optional	Mandatory	See SubjectPublicKeyInfo defined in Table 10
issuerUID		Must be omitted	Shall not be used	
subjectUID		Must be omitted	Shall not be used	
extensions	Extensions	Optional	May be used	

Table 9. CertTemplate

SubjectPublicKeyInfo (RFC-5280)				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
algorithm	AlgorithmIdentifier	Mandatory	Mandatory	The default AlgorithmIdentifier shall be id-ecPublicKey with parameter secp256r1 ([RFC-5480]) For backwards compatibility reasons. (e.g., for the KM use case), the AlgorithmIdentifier rsaEncryption shall be supported.
subjectPublicKey		Mandatory	Mandatory	Bytes of public key to be signed (depends on algorithm used)

Table 10. SubjectPublicKeyInfo

POPOSigningKey (see RFC-4211**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
poposkInput		Optional	Shall not be used	
algorithmIdentifier	AlgorithmIdentifier	Mandatory	Mandatory	The algorithmIdentifier shall be ecdsa-with-SHA256 or ecdsa-with-SHA512 For KM use case, the algorithmIdentifier shall be sha384WithRSAEncryption ([RFC-4055])
signature		Mandatory	Mandatory	Signature (depends on signing algorithm used)

Table 11. POPOSigningKey

5.5.3.4 CMP message specification: “Certificate Response” message

5.5.3.4.1 For the “Certificate Response” message, the body of the PKI message shall comply with Table 12, Table 13, Table 14, Table 15, Table 16 and Table 17.

CertRepMessage (see RFC-4210**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
caPubs 1..MAX	Sequence of Cetificates	Optional	Mandatory	CA public keys
response	CertResponse	Mandatory	Mandatory	See CertResponse in Table 12

Table 12. CertRepMessage

CertResponse (see RFC-4210**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
certReqId	INTEGER	Mandatory	Mandatory	Must be same as for corresponding request

status	PKIStatusInfo	Mandatory	Mandatory	
certifiedKeyPair	CertifiedKeyPair	Optional	Mandatory	See CertifiedKeyPair defined in Table 13
rspInfo		Optional	Shall not be used	

Table 13. CertResponse

CertifiedKeyPair (see RFC-4210**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
certOrEncCert	CertOrEncCert	Mandatory		See CertOrEncCert in Table 14
privateKey		Optional	Shall not be used	
publicationInfo		Optional	Shall not be used	

Table 14. CertifiedKeyPair

CertOrEncCert (see RFC-4210**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
certificate or encryptedCert		Mandatory	Mandatory	Certificate shall be chosen

Table 15. CertOrEncCert

Certificate (see RFC-5280**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
tbsCertificate	TBSCertificate	Mandatory	Mandatory	See TBSCertificate defined in Table 16
signatureAlgorithm	AlgorithmIdentifier	Mandatory	Mandatory	The algorithmIdentifier shall be ecdsa-with-SHA256 For KM use case, the algorithmIdentifier shall be sha384WithRSAEncryption
signatureValue	BIT STRING	Mandatory	Mandatory	Digital signature of tbsCertificate

Table 16. Certificate

TBSCertificate (see RFC-5280**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples

version		INTEGER	Mandatory	Fixed value	Must be set to “2”
serialNumber		INTEGER	Mandatory	Mandatory	Unique for each certificate
signature		AlgorithmIdentifier	Mandatory	Mandatory	The algorithmIdentifier shall be ecdsa-with-SHA256 For KM use case, the algorithmIdentifier shall be sha384WithRSAEncryption
issuer		Distinguished Name	Mandatory	Mandatory	DN of CA, see Table 19
validity		Sequence of two dates (notBefore, notAfter)	Mandatory	Mandatory	UTC time shall used for notBefore and notAfter
subject		Distingushed Name	Mandatory	Mandatory	DN of requester, see Table 19
subjectPublicKeyInfo		AlgorithmIdentifier	Mandatory	Mandatory	See SubjectPublicKeyInfo defined in Table 9, must be same as in request for supplied public key
issuerUniqueID			Optional	Shall not be used	
subjectUniqueID			Optional	Shall not be used	
Certificate extensions	Authority Key Identifier	Sequence	Mandatory for not « self-signed » conforming CA.	Mandatory	
	Subject Key Identifier	OCTET STRING	Mandatory for conforming CA	Mandatory	
	Key Usage	BIT STRING	Optional	Mandatory	Key usage shall indicate at least that certificate MUST allow the key to be used for signing and additionally also keyEncipherment If in Table 10 SubjectPublicKeyInfo AlgorithmIdentifier value is set to " id-ecPublicKey”, the Key usage shall not indicate keyEncipherment or dataEncipherment [according RFC 8813].
	Certificate Policies		Optional	Shall not be used	
	Policy Mappings		Optional	Shall not be used	
	Subject Alternative Name		Optional	May be used	The common name according to Table 19 shall be used.
	Issuer Alternative Name		Optional	Shall not be used	
	Subject Directory Attributes		Optional	Shall not be used	

	Basic Constraints		Conforming CAs MUST include this extension in all CA certificates that contain public keys used to validate digital signatures on certificates and MUST mark the extension as critical in such certificates.	Mandatory	Critical: yes, CA: false
	Name Constraints		Optional	Shall not be used	
	Policy Constraints		Optional	Shall not be used	
	Extended Key Usage		Optional	May be used.	Should be set to clientAuth for Onboard devices (OBU, ATO-OB). Should be set to serverAuth for trackside devices (RBC, ATO-TS, KM, PKI).
	CRL Distribution Points	Sequence of DistributionPoint	Optional	Mandatory	Identifies the CRL distribution, i.e., the URI of the CRL. Not used for KMS use case
	Inhibit anyPolicy		Optional	Shall not be used	
	Freshest CRL		Optional	Shall not be used	
	Authority Information Access	Sequence of AccessDescription	Optional	Mandatory	Used for KM use case only: Identity of the URI (FQDN) of the OCSP responder.
	Subject Information Access		Optional	Shall not be used	

Table 17. TBSCertificate

5.5.3.5 CMP message specification: “Certification Confirmation” message

5.5.3.5.1 For the “Certificate Confirmation” message, the body of the PKI message shall comply with Table 18.

CertConfirmContent (see RFC-4210**)**				
Field name	Data type	Req. by standard	Req. by subset	Notes / Examples
certHash	OCTECT STRING	Mandatory	Mandatory	
certReqId	INTEGER	Mandatory	Mandatory	Must match corresponding CertRespMessage
statusInfo		Optional	Shall not be used	

Table 18. CertConfirmContent

5.5.3.6 CMP message specification: “Confirmation Acknowledgement” message

5.5.3.6.1 For the “Confirmation Acknowledgement” message, the body of the PKI message shall be empty (see PKIConfirmContent in RFC-4210).

5.5.4 Distinguished Name

5.5.4.1 A Distinguished Name is a name given to an element within a computer system or a network that uniquely identifies it.

5.5.4.2 The Distinguished Name syntax is defined in standards [X.520], [X.500] and [X.501].

5.5.4.3 A Distinguished Name is made up of “attribute=value” pairs, separated by commas.

5.5.4.4 The PKI user shall use names with the attributes in the order stated here below:

Distinguished Name		
Key identifier	Attribute type	Content
C	Country Code	ISO alpha-2 country code
O	Organization Name	Acronym of the organisation operating the element identified by the OID. This acronym shall be composed of 2 or 3 uppercase characters from the Latin alphabet [ISO-8859-1].
OU	Organizational Unit Name	Element abbreviation shall be used as Unit name. I.e. one of the following: • KMC • RBC • EVC • RIU • ATO-OB • ATO-TS Note: The element corresponds to the ETCS ID type defined in [Subset-037-1].
CN	Common Name	The common name represents the name given to the element. The Common Name shall be a fully qualified domain name (FQDN). The format of the FQDN shall be “id<ETCS ID>.ty<ETCS-ID Type>.etcs”, as defined in [Subset-037-1].
serialNumber	Serial Number	ETCS ID

Table 19. Distinguished Name Syntax

5.6 Certificate status check interface

5.6.1 General

5.6.1.1 The status of a certificate shall be checked for validity.

5.6.2 Certificate status check functions

5.6.2.1 The following table provides the functions allocation for the certificate status check interface:

Function	Allocation	Purpose
CRL Download	PKI Client	Request for a certificate revocation list
CRL Publication	PKI Server	Provide certificate revocation list
OCSP Request	PKI client	Request for the revocation status of a certificate
OCSP Response	PKI server	Provide the revocation status of a certificate

Note: OCSP is only used for the KM use case.

5.6.2.2 Certificate status check using CRLs

5.6.2.2.1 The PKI client shall check whether the certificate of a peer has been revoked, by using the respective CRL, see [RFC-5280].

5.6.2.2.2 The PKI server shall publish CRLs, see [RFC-5280].

5.6.2.2.3 The PKI server shall update respective CRLs immediately, once the status of a certificate in the CRL changed.

5.6.2.2.4 The PKI client shall check for and retrieve updated CRLs from the PKI server at least once every 24 hours.

5.6.2.3 Certificate status check using OCSP

5.6.2.3.1 The PKI client may check whether the certificate of a peer has been revoked, by sending an OCSP Request to the OCSP responder.

5.6.2.3.2 The OCSP responder shall send an OCSP Response to a PKI client having received an OCSP Request.

5.6.3 Interface specification

5.6.3.1 General requirements

5.6.3.1.1 Certificate status information shall be provided by using certificate revocation lists (CRLs), see [RFC-5280]. Certificate status information may be additionally provided by using the Online Certificate Status Protocol (OCSP), see [RFC-6960].

5.6.3.2 CRL requirements

5.6.3.2.1 CRLs format and syntax shall comply with X.509 v2 CRL, see [RFC-5280].

5.6.3.2.2 The HTTP protocol shall be used for download of CRLs.

5.6.3.3 OCSP Requirements

5.6.3.3.1 The OCSP Request and OCSP Response shall be exchanged in the same session.

5.6.3.3.2 The HTTP protocol shall be used for transfer of OCSP messages, see [RFC-6960].

5.6.3.3.3 In case the OCSP certificate check process is completed with OCSP status either *good* or *revoked*, the OCSP certificate check process shall be considered as successful.

5.6.3.3.4 In case the process is not completed or completed with OCSP response status "*unknown"* during the certificate check, the OCSP certificate check process shall be considered as failed.

5.6.3.3.5 In case of a failed OCSP certificate check process, respective CRLs shall be evaluated instead.

5.6.3.3.6 The OCSP Request and OCSP Response messages shall conform to OCSP protocol described in [RFC-6960].

5.6.3.3.7 Note: For each optional field in the OCSP messages (see Table 20 to Table 28), it is stated if this optional field:

- shall be used
- shall not be used
- may be used

5.6.3.3.8 The check of the peer certificate chain shall be performed at the reception of any certificate from a peer entity unless this information is provided by the "Multiple Certificate Status Request" extension.

5.6.3.3.9 The PKI server shall send an OCSP Response if these two conditions are fulfilled:

a) The OCSP Request message is compliant to [RFC-6960]

b) The request contains the information needed by the OCSP server

5.6.3.3.10 OCSP message specification: OCSP Request

5.6.3.3.10.1 The OCSP Request shall comply with Table 20, Table 21, Table 22 and Table 23.

OCSPRequest (see RFC-6960)		
Field	Cardinality	Applicability
tbsRequest	Mandatory	See TBSRequest in Table 21
optionalSignature	Optional	Shall not be used

Table 20. OCSPRequest

TBSRequest (see RFC-6960)		
Field	Cardinality	Applicability
version	Mandatory	
requestorName	Optional	Shall not be used
requestList	Mandatory	See Request defined in Table 22
requestExtensions	Optional	Shall not be used

Table 21. TBSRequest

Request (see RFC-6960)		
Field	Cardinality	Applicability
reqCert	Mandatory	See CertID defined in Table 23
singleRequestExtensions	Optional	Shall not be used

Table 22. Request

CertID (see RFC-6960)		
Field	Cardinality	Applicability
hashAlgorithm	Mandatory	The AlgorithmIdentifier shall be SHA1.
issuerNameHash	Mandatory	
issuerKeyHash	Mandatory	
serialNumber	Mandatory	

Table 23. CertID

5.6.3.3.11 OCSP message specification: OCSP Response

5.6.3.3.11.1 The OCSP Response shall comply with Table 24, Table 25, Table 26, Table 27 and Table 28.

OCSPResponse (see RFC-6960)		
Field	Cardinality	Applicability
responseStatus	Mandatory	
responseBytes	Optional	Shall be used See responseBytes defined in Table 25

Table 24. OCSPResponse

ResponseBytes (see RFC-6960)		
Field	Cardinality	Applicability
responseType	Mandatory	Shall be id-pkix-ocsp-basic
response	Mandatory	Shall be the DER encoding (see [X.690]) of BasicOCSPResponse defined in Table 26

Table 25. ResponseBytes

BasicOCSPResponse (see RFC-6960)		
Field	Cardinality	Applicability
tbsResponseData	Mandatory	See ResponseData defined in Table 27
signatureAlgorithm	Mandatory	The AlgorithmIdentifier shall be sha384WithRSAEncryption
signature	Mandatory	
certs	Optional	Shall be used

Table 26. BasicOCSPResponse

ResponseData (see RFC-6960)		
Field	Cardinality	Applicability
version	Mandatory	
responderID	Mandatory	
producedAt	Mandatory	
responses	Mandatory	See SingleResponse defined in Table 28
responseExtensions	Optional	Shall not be used

Table 27. ResponseData

SingleResponse (see RFC-6960)		
Field	Cardinality	Applicability
certID	Mandatory	See CertID defined in Table 23
certStatus	Mandatory	
thisUpdate	Mandatory	
nextUpdate	Optional	Shall not be used
singleExtensions	Optional	Shall not be used

Table 28. SingleResponse

5.7 Requirements on the Public Key Infrastructure

5.7.1.1 The security certificate of an ERTMS application shall be managed via a Public Key Infrastructure (PKI) typically belonging to an infrastructure manager, or the rail operator related to (or owning) the engine/locomotive hosting the ERTMS application. Similar for the trackside equipment (RBC, KMC etc.)

5.7.1.2 The CA/RA function of the PKI shall manage creation and renewal of security certificates for all its related ERTMS applications.

5.7.1.3 The CA/RA function of the PKI shall manage revocation of security certificates for all its related ERTMS applications.
Note: For example, if the private key can be suspected of being compromised or by other similar reasons.

5.7.1.4 The relevant PKI according to the use cases shall be accessible via any network, i.e. for,

- Enrolment and renewal of certificates
- Certificate Revocation List (CRL) download
- Certificate validation via OCSP

5.7.1.5 The CRL shall be published periodically every 24 hours or less or may be published immediately after a certificate has been revoked.

5.8 Intentionally Deleted

5.9 Supervision and Diagnostics

5.9.1 Fail Secure Principle

5.9.1.1 The End-to-End Security shall follow the fail secure principle.
Note: Fail secure is defined as “the security function or the secure system delivering the function remains in the secure state.” [50701.C.4]

5.9.1.2 The fail secure principle shall be followed only in case no safety requirement is undermined or contradicted. In all other cases safety requirements and architecture should be dominant. [50701.C.4]

5.9.1.3 The End-to-End Security shall prevent any communication when there is a failure in the TLS communication (Fail Close / deterministic output).
Note: The End-to-End Security relies on central security services (e.g., PKI and Time) or other central network services (as e.g., FRMCS). Secure operation in degraded modes of central services is not addressed by this standard.

5.9.2 Error Reporting

5.9.2.1 All diagnostic information and alarms raised by End-to-End Security shall be logged by the endpoint application (onboard and trackside).

5.9.2.2 The TLS user shall send an error reason defined in chapter 5.9.2.8 to the endpoint application, if TLS connection may not be established or will change to down.

5.9.2.3 Diagnostic information shall be provided from any relevant OSI-Layer.
Note: TLS connection error are caused on OSI-Layer 5. An application (TLS – Endpoint) typical has no access to error information, reported from lower OSI-Layer.

5.9.2.4 The logging shall be based on syslog (RFC-5424).

5.9.2.5 The transport of the syslog messages shall be protected by TLS [RFC-5425].

5.9.2.6 The provided diagnostic information shall be archived locally persistent by application for future analysis.
Note: Persistency shall allow to hold buffer for 90 days.

5.9.2.7 To support trouble shooting within a distributed security system (such as PKI) diagnoses and alarms shall be reported classified according Table 29 TLS Error Reason Code
Note: Classified logging standard is vital to support security maintenance and thus interoperation as such.
Note: The diagnostic information interface to application is subject of design and will not be defined by this specification.
Note: TLS error codes are subject of underlaying TLS Libraries (which are a choice of design). Nevertheless, for a standardized Logging, specific error codes shall be mapped to most common used [OpenSSL] values (see Table 30 up-to Table 38).
Note: upcoming centralized remote logging will also depend on classified error reporting.
Note: Detailed information about error codes can be found by [x509errors].

5.9.2.8 Error Reason Codes
TLS error can be classified by reason code (see following blow). Each class is refined by sub-reason codes. Error Logging shall contain at least calling application, address- and session information, time stamp, reason code and sub-reason code.

Reason Code (Dec)	Description
0	No Error - Operation successful
1	Time validity errors
2	Trust or chain related errors
3	Basic extension errors
4	Name related errors
5	Usage and policy errors
6	Algorithm related errors
7	Formatting errors
8	Uncategorized errors

Table 29 TLS Error Reason Code

5.9.2.9 No Error - Operation successful
This class is used, when operation was successful.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
0	0	X509_V_OK	The operation was successful.

Table 30 TLS No Error - Operation successful

5.9.2.10 Time validity errors

Errors occurring when a certificate is outside its validity period or when it is revoked by its CA.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
1	9	X509_V_ERR_CERT_NOT_YET_VALID	The certificate is not yet valid: the notBefore date is after the current time.
1	10	X509_V_ERR_CERT_HAS_EXPIRED	The certificate has expired (its validity period passed).
1	11	X509_V_ERR_CRL_NOT_YET_VALID	The CRL is not yet valid.
1	12	X509_V_ERR_CRL_HAS_EXPIRED	The CRL has expired.
1	23	X509_V_ERR_CERT_REVOKED	The certificate has been revoked.

Table 31 Time validity errors

5.9.2.11 Trust or chain related errors

These errors occur when the trust chain to the root certificate is not built correctly or fails.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
2	2	X509_V_ERR_UNABLE_TO_GET_ISSUER_CERT	The issuer certificate of a looked-up certificate could not be found. This normally means the list of trusted certificates is not complete.
2	3	X509_V_ERR_UNABLE_TO_GET_CRL	The CRL of a certificate could not be found.
2	18	X509_V_ERR_DEPTH_ZERO_SELF_SIGNED_CERT	The provided certificate is self-signed, and it is not present in the list of trusted certificates.
2	19	X509_V_ERR_SELF_SIGNED_CERT_IN_CHAIN	The certificate chain could be built up using the untrusted certificates, but the root could not be found locally.
2	20	X509_V_ERR_UNABLE_TO_GET_ISSUER_CERT_LOCALLY	The issuer certificate could not be found this occurs if the issuer certificate of an untrusted certificate cannot be found.
2	21	X509_V_ERR_UNABLE_TO_VERIFY_LEAF_SIGNATURE	No signatures could be verified because the chain contains only one certificate, and it is not self-signed.
2	22	X509_V_ERR_CERT_CHAIN_TOO_LONG	The certificate chain length is greater than the supplied maximum depth. Unused.
2	27	X509_V_ERR_CERT_UNTRUSTED	The root CA is not marked as trusted for the specified purpose.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
2	29	X509_V_-ERR_SUBJECT_ISSUER_MISMATCH	The current candidate issuer certificate was rejected because its subject name did not match the issuer name of the current certificate.
2	30	X509_V_ERR_AKID_SKID_MISMATCH	The current candidate issuer certificate was rejected because its subject key identifier was present and did not match the authority key identifier current certificate.
2	31	X509_V_ERR_AKID_ISSUER_SERIAL_MISMATCH	The current candidate issuer certificate was rejected because its issuer name and serial number was present and did not match the authority key identifier of the current certificate.
2	33	X509_V_ERR_UNABLE_TO_GET_CRL_ISSUER	Unable to get CRL issuer certificate.
2	44	X509_V_ERR_DIFFERENT_CRL_SCOPE	Different CRL scope.
2	54	X509_V_ERR_CRL_PATH_VALIDATION_ERROR	CRL path validation error.
2	55	X509_V_ERR_PATH_LOOP	Path loop.
2	75	X509_V_ERR_OCSP_CERT_UNKNOWN	Returned by the verify call back to indicate that the certificate is not recognized by the OCSP responder.
2	74	X509_V_ERR_OCSP_VERIFY_FAILED	Returned by the verify call back to indicate OCSP verification failed.

Table 32 Trust or chain related errors

5.9.2.12 Basic extension errors

Errors related to extensions in general or to the BasicConstraints standard extension.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
3	24	X509_V_ERR_INVALID_CA	A CA certificate is invalid. Either it is not a CA, or its extensions are not consistent with the supplied purpose
3	25	X509_V_ERR_PATH_LENGTH_EXCEEDED	The allowed length of the certification path was exceeded.
3	34	X509_V_ERR_UNHANDLED_CRITICAL_EXTENSION	A critical extension was not recognized or could not be processed.
3	36	X509_V_ERR_UNHANDLED_CRITICAL_CRL_EXTENSION	Unhandled critical CRL extension.
3	41	X509_V_ERR_INVALID_EXTENSION	Invalid or inconsistent certificate extension.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
3	45	X509_V_ERR_UNSUPPORTED_EXTENSION_FEATURE	Unsupported extension feature.

Table 33 Basic extension errors

5.9.2.13 Name related errors

Errors occurring when a certificate is outside its validity period or when it is revoked by its CA.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
4	46	X509_V_ERR_UNNESTED_RESOURCE	RFC-3779 resource not subset of parent's resources.
4	47	X509_V_ERR_PERMITTED_VIOLATION	Permitted subtree violation.
4	48	X509_V_ERR_EXCLUDED_VIOLATION	Excluded subtree violation.
4	49	X509_V_ERR_SUBTREE_MINMAX	Name constraints minimum and maximum not supported.
4	51	X509_V_ERR_UNSUPPORTED_CONSTRAINT_TYPE	Unsupported name constraint type.
4	52	X509_V_ERR_UNSUPPORTED_CONSTRAINT_SYNTAX	Unsupported or invalid name constraint syntax.
4	53	X509_V_ERR_UNSUPPORTED_NAME_SYNTAX	unsupported or invalid name syntax
4	62	X509_V_ERR_HOSTNAME_MISMATCH	The requested hostname does not match the subject name in the certificate.
4	64	X509_V_ERR_IP_ADDRESS_MISMATCH	IP address mismatch.

Table 34 Name related errors

5.9.2.14 Usage and policy errors

Errors related to standard extensions CertificatePolicies, KeyUsage and ExtendedKeyUsage.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
5	26	X509_V_ERR_INVALID_PURPOSE	The certificate’s key is being used for a different purpose than allowed.
5	28	X509_V_ERR_CERT_REJECTED	The root CA is marked to reject the specified purpose.
5	32	X509_V_ERR_KEYUSAGE_NO_CERTSIGN	The current candidate issuer certificate was rejected because its keyUsage extension does not permit certificate signing.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
5	35	X509_V_ERR_KEYUSAGE_NO_CRL_SIGN	Key usage does not include CRL signing.
5	39	X509_V_ERR_KEYUSAGE_NO_DIGITAL_SIGNATURE	Key usage does not include digital signature.
5	42	X509_V_ERR_INVALID_POLICY_EXTENSION	Invalid or inconsistent certificate policy extension.
5	43	X509_V_ERR_NO_EXPLICIT_POLICY	No explicit policy.

Table 35 Usage and policy errors

5.9.2.15 Algorithm related errors

Various errors signalizing usage of invalid or deprecated algorithms.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
6	56	X509_V_ERR_SUITE_B_INVALID_VERSION	Suite B: certificate version invalid
6	57	X509_V_ERR_SUITE_B_INVALID_ALGORITHM	Suite B: invalid public key algorithm.
6	58	X509_V_-ERR_SUITE_B_INVALID_CURVE	Suite B: invalid ECC curve
6	59	X509_V_ERR_SUITE_B_INVALID_SIGNATURE_-ALGORITHM	Suite B: invalid signature algorithm.
6	60	X509_V_ERR_SUITE_B_LOS_NOT_ALLOWED	Suite B: curve not allowed for this LOS
6	61	X509_V_ERR_SUITE_B_CANNOT_SIGN_P_384_WITH_P_256	Suite B: cannot sign P-384 with P-256.
6	66	X509_V_ERR_EE_KEY_TOO_SMALL	EE certificate key too weak.
6	67	X509_V_ERR_CA_KEY_TOO_SMALL	CA certificate key too weak.
6	68	X509_V_ERR_CA_MD_TOO_WEAK	CA signature digest algorithm too weak.
6	76	X509_V_ERR_UNSUPPORTED_SIGNATURE_ALGORITHM	Cannot find certificate signature algorithm
6	78	X509_V_ERR_SIGNATURE_ALGORITHM_INCONSISTENCY	CERT info signature and signature algorithm mismatch

Table 36 Algorithm related errors

5.9.2.16 Formatting errors

These errors occur when a field of the certificate/CRL contains invalid values or is badly formatted.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
7	4	X509_V_ERR_UNABLE_TO_DECRYPT_CERT_SIGNATURE	The certificate signature could not be decrypted. This means that the actual signature value could not be determined rather than it not matching the expected value, this is only meaningful for RSA keys.
7	6	X509_V_ERR_UNABLE_TO_DECODE_ISSUER_PUBLIC_KEY	The public key in the certificate SubjectPublicKeyInfo could not be read.
7	7	X509_V_ERR_CERT_SIGNATURE_FAILURE	The signature of the certificate is invalid.
7	8	X509_V_ERR_CRL_SIGNATURE_FAILURE	The signature of the certificate is invalid. (
7	13	X509_V_ERR_ERROR_IN_CERT_NOT_BEFORE_FIELD	The certificate notBefore field contains an invalid time.
7	14	X509_V_ERR_ERROR_IN_CERT_NOT_AFTER_FIELD	The certificate notAfter field contains an invalid time.
7	15	X509_V_ERR_ERROR_IN_CRL_LAST_UPDATE_FIELD	The CRL lastUpdate field contains an invalid time.
7	16	X509_V_-ERR_ERROR_IN_CRL_NEXT_UPDATE_FIELD	The CRL nextUpdate field contains an invalid time.
7	5	X509_V_ERR_UNABLE_TO_DECRYPT_CRL_SIGNATURE	The CRL signature could not be decrypted: this means that the actual signature value could not be determined rather than it not matching the expected value. Unused.
7	24	X509_V_ERR_NO_ISSUER_PUBLIC_KEY	Issuer certificate doesn't have a public key.
7	77	X509_V_ERR_SIGNATURE_ALGORITHM_MISMATCH	Subject signature algorithm and issuer public key algorithm mismatch
7	80	X509_V_ERR_PATHLEN_INVALID_FOR_NON_CA	Path length invalid for non-CA cert
7	81	X509_V_ERR_PATHLEN_WITHOUT_KU_KEY_CERT_SIGN	Path length given without key usage keyCertSign
7	82	X509_V_ERR_KU_KEY_CERT_SIGN_INVALID_FOR_NON_CA	Key usage keyCertSign invalid for non-CA cert
7	83	X509_V_ERR_ISSUER_NAME_EMPTY	Issuer name empty
7	84	X509_V_ERR_SUBJECT_NAME_EMPTY	Subject name empty
7	85	X509_V_ERR_MISSING_AUTHORITY_KEY_IDENTIFIER	Missing Authority Key Identifier
7	86	X509_V_ERR_MISSING_SUBJECT_KEY_IDENTIFIER	Missing Subject Key Identifier
7	87	X509_V_ERR_EMPTY_SUBJECT_ALT_NAME	Empty Subject Alternative Name extension

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
7	88	X509_V_ERR_EMPTY_SUBJECT_SAN_NOT_CRITICAL	Subject empty and Subject Alt Name extension not critical
7	89	X509_V_ERR_CA_BCONS_NOT_CRITICAL	Basic Constraints of CA cert not marked critical
7	90	X509_V_ERR_AUTHORITY_KEY_IDENTIFIER_CRITICAL	Authority Key Identifier marked critical
7	91	509_V_ERR_SUBJECT_KEY_IDENTIFIER_CRITICAL	Subject Key Identifier marked critical
7	92	X509_V_ERR_CA_CERT_MISSING_KEY_USAGE	CA cert does not include key usage extension
7	93	X509_V_ERR_EXTENSIONS_REQUIRE_VERSION_3	Using cert extension requires at least X509v3 (source
7	94	X509_V_ERR_EC_KEY_EXPLICIT_PARAMS	Certificate public key has explicit ECC parameters

Table 37 Formatting errors

5.9.2.17 Uncategorized errors

These errors are not yet categorized, deprecated or not used at all.

Reason Code (Dec)	Sub-reason Code (Dec)	Code	Description
8	1	X509_V_ERR_UNSPECIFIED	Unspecified error; should not happen.
8	17	X509_V_ERR_OUT_OF_MEM	An error occurred trying to allocate memory.
8	37	X509_V_ERR_INVALID_NON_CA	Invalid non-CA certificate has CA markings.
8	38	X509_V_ERR_PROXY_PATH_LENGTH_EXCEEDED	Proxy path length constraint exceeded.
8	40	X509_V_ERR_PROXY_CERTIFICATES_NOT_ALLOWED	Proxy certificates not allowed, please use -allow_proxy_certs.
8	65	X509_V_ERR_DANE_NO_MATCH	DANE TLSA authentication is enabled, but no TLSA records matched the certificate chain.
8	69	X509_V_ERR_INVALID_CALL	Invalid certificate verification context.
8	70	X509_V_ERR_STORE_LOOKUP	Issuer certificate lookup error.
8	72	X509_V_ERR_PROXY_SUBJECT_NAME_VIOLATION	Proxy certificate name violation.
8	71	X509_V_ERR_NO_VALID_SCTS	Certificate Transparency required, but no valid SCTs found.
8	73	X509_V_ERR_OCSP_VERIFY_NEEDED	Returned by the verify call-back to indicate an OCSP verification is needed.

Table 38 Uncategorized errors

5.9.3 Error Handling

5.9.3.1 TLS error handling in context of session establishment or protocol violation shall be managed by the TLS protocol stack.

5.9.3.2 In any case a faulty certificate shall prevent TLS connection establishment. See also 5.4.2.6.3.

5.9.3.3 Certificate validation failure shall prevent TLS connection establishment.
Note: Certificate validation is subject of 5.9.1 Fail Secure Principle: Security principle is in case of findings to keep connection availability as long as security is not compromised. In particular: a CRL time out shall raise an alarm only.

5.9.3.4 It is recommended that a CRL should be valid for 1 month and being renewed once per day.

5.9.3.5 Errors shall be reported according to 5.9.2 Error Reporting.

ANNEX A. ERTMS Application Cross Reference

A.1. General Information

A.1.1.1 The sections below show which requirements that are applicable to each ERTMS application.

A.2. On-line Key Management

A.2.1.1 Requirements from chapter 5.2 to 5.9 are applicable to On-line Key Management [Subset-137], with the exceptions and distinctions specified in Table 39 below.

Section	Comment
5.4	KMC server shall support both TLS 1.2 & 1.3 for backwards compatibility.
5.4.1	The KMC, OBU and RBC shall implement the following TLS features, KMC – TLS client and TLS server OBU – TLS client RBC – TLS server
5.4.2.3	Safety keys are confidential, thus, only cipher suites that encrypt the application payload shall be used.
5.5.1.2	FQDN resolution shall be performed using DNS.
5.5.2.3.11	The characters used for the passphrase shall be encoded using UTF-8 with a minimum length of 16 characters.
5.6	OCSP shall be used to check certificate revocation (CRL’s are not used).

Table 39. On-line Key Management

A.3. ATO End-to-End Security

A.3.1.1 Requirements from chapter 5.2. to 5.9 are applicable to Automatic Train Operation [Subset-148], with the exceptions and distinctions specified in Table 40 below.

Section	Comment
5.4	Application payload is non-confidential; thus, all cipher suites can be used.
5.4.1	The ATO-OB and ATO-TS shall implement the following TLS features, TLS-OB – TLS client TLS-TS – TLS server
5.4.3	Pre-shared keys shall not be used.
5.5.1.2	FQDN resolution shall be performed using DNS.

5.6	CRL shall be used to check certificate revocation (OCSP is not used).

Table 40. ATO End-to-End Security

A.4. ETCS over FRMCS

A.4.1.1 Requirements from chapter 5.2. to 5.9 are applicable to Automatic Train Protection [Subset-037-3], with the exceptions and distinctions specified in Table 41 below.

Section	Comment
5.4	Application payload is non-confidential; thus, all cipher suites can be used.
5.4.1	The OBU and RBC shall implement the following TLS features, OBU – TLS client RBC – TLS server
5.4.3	Pre-shared keys shall not be used.
5.5.1.2	FQDN resolution shall be performed using DNS.
5.6	CRL shall be used to check certificate revocation (OCSP is not used).

Table 41. ATP End-to-End Security