ERTMS/ETCS
Off-line Key Management FIS
REF : SUBSET-038 ISSUE : 3.1.0 DATE : 17-12-15

Company	Technical Approval	Management approval
ALSTOM		
ANSALDO		
AZD		
BOMBARDIER		
CAF		
SIEMENS		
THALES		

1. MODIFICATION HISTORY

Issue Number Date	Section Number	Modification / Description	Author
2.2.1 28-09-09	All	Result from splitting of document 05E537-1C (technical part)	F. Hausman
2.2.2 17-03-10	All § 6.5.1.3, tables 7 to 13	Wording check after splitting. CR 749	WG
2.2.3 26-05-10	All	Update of revision marks : only CR 749 is traced	WG
2.2.4 04-11-10	Tables 7 to 13 Table 6, foot note page 31 § 4.1.1.1, 6.2.1.2, 3.2.1.4, 6.1.1.4, 6.3.1.1, 8.1.1.2, 8.3.1.3.1, 8.3.2.3, 5 and tables 2, 7 to 13	Implementation of CR 749 as agreed on 24-09-08. Consistency check ERA comments	WG
2.2.5 05-11-10	Footer update, §7.1.1.1	SC comments	WG
2.3.0	Release version number	Agreed at ERA Control Group 11.11.10 with no changes other than release version number	D. Gillan
2.3.1 13-01-12	§ 6.4.1, 6.4.2, 4.1, 8.3.1.3.1, tables 7 to table 13 § 8.3, 4.1 § 7.2.6, 7.2.7, 7.2.8, 7.2.10.2, 8.1.1.1, 8.4.3.1, 8.4.3.4, 8.4.4.1, 8.4.4.4 + tables 6 and 8 to 14	CR 970 CR 971 CR 1132	WG
2.3.2 18-01-12	§ 3.2.1.2, 5, 6.2.2.1, 6.2.3.1, 6.2.3.2,	CR 1132	WG

	7.2.1.1, 7.2.2, 8.3.2.4, 8.3.1.3, 8.4.1.3, 4.2		
2.3.3 20-01-12	See § impacted in issue 2.3.2	CR 1132	WG
2.3.4 15-02-12	Figure 2, § 5	Update following EEIG remark on release 2.3.3.	WG
2.3.5 24-02-12	§ 1, 3.1.1.2, 4.2 and front page	CR 1139	WG
3.0.0 28-02-12	§1 and front page	Baseline 3 release version	WG
3.0.1 15-12-15	§ 5	CR 1237	FH/PP
3.1.0 17-12-15	-	Baseline 3 2nd release version	PP

2. TABLE OF CONTENTS

3. INTRODUCTION

3.1 Subject

3.1.1.1 No operational intervention is normally necessary to allow a duly authorised OBU to traverse into several separately controlled ERTMS areas, provided that the relevant preparatory actions have been carried out in advance of arrival at each area. Among these actions specific KM functions are also required to establish interoperable services.

3.1.1.2 For KMC interworking, a FIS (Functional Interface Specification) is required at the KMC-KMC interface. This is the purpose of this document, which describes the principles and procedures including the use of additional K-KMC keys required to exchange KMACs and allow interoperable train traffic between KM domains. It also describes general concepts, principles, functions and procedures to manage cryptographic materials used by the EuroRadio safety layer for ERTMS/ETCS applications.

3.1.1.3 This specification covers management of Symmetric Key Material using symmetric procedures.

3.2 Field Of Application

3.2.1.1 This document is related to the key management of the following ERTMS entities:

- OBU
- RIU
- RBC
- KMC

3.2.1.2 It contains the necessary pre-conditions for key exchange between KMCs.

3.2.1.3 The document covers Off-line exchange needed between KM domains to manage different sets of ERTMS entities and allow railway traffic running in a fully controlled and interoperable way.

3.2.1.4 The document does not cover key management inside a KM domain for trackside or OBU ERTMS entities.

3.2.1.5 Fulfilling the requirements stated in this document contributes to achieving the security necessary for the protection of an ERTMS application against malicious intrusions. This document defines neither the security targets nor all additional measures, e.g. a System Security Policy, that may be necessary to achieve them.

3.2.1.6 The High Level Operational Rules with regards to the interoperability aspects for the implementation and operation of the Key Management Systems are specified in Ref. [8].

3.3 Document Description

3.3.1.1 This document contains eight chapters.

3.3.1.2 Chapter 6 provides general KM concepts and principles.

3.3.1.3 Chapter 7 defines basic KM roles and functions.

3.3.1.4 Chapter 8 contains the off-line KM transactions, and constitutes the core requirements of this FIS.

4. REFERENCES

4.1 Normative References

4.1.1.1 This FIS incorporates by dated or undated references, provisions from other publications. These normative references are cited at the appropriate place in the text and the publications are listed hereafter. For dated references, subsequent amendments to or revisions of any of these publications apply to this FIS only when incorporated in it by amendment or revision. For undated references the latest edition of the publication referred to applies.

[1] UNISIG EURORADIO FIS Subset 037

[2] UNISIG KMC-ETCS Entity Offline KM FIS Subset 114

[3] ANSI X9.52 – 1998 Triple Data Encryption Algorithm Modes of Operation (parts only)

[4] ANSI X3.92 – 1981 Data Encryption Standard (DES) algorithm

4.2 Informative References

[5] EEIG ERTMS/ETCS Key Management System URS – ref 02E189 v1 - 14/05/02

[6] EEIG ETCS Key Management System FRS – ref 02E266 v1 - 17/12/02

[7] UNISIG Glossary of Terms and Abbreviations Subset-023

[8] EEIG KMS Operational Aspects-04E518-0N

[9] UNISIG RBC-RBC Safe Communication Interface Subset-098

5. ABBREVIATIONS AND DEFINITIONS

CBC	Cipher Block Chaining
DES	Data Encryption Standard (ANSI X3.92-1981)
K-KMC	Inter KMC key composed of two parts K-KMC1 and K-KMC2
KSMAC	Session Key
Triple-DES	Triple-Data Encryption Standard

Refer to the UNISIG Glossary [7] for other definitions and abbreviations.

Confidentiality	Used to prevent information from being read by unauthorised entities.
Crypto period	Time up to which it is anticipated that the cryptographic algorithms and/or the hardware / software meet the required security targets.
Cryptography	The discipline which embodies the principles, means and methods for the transformation of data in order to hide its information content, prevent its undetected modification, prevent its unauthorised use or a combination thereof.
Domain	One domain is defined by one KMC (Home KMC) and all the on-board and trackside entities using that KMC for key management purposes. Note : a domain may contain only on-board or trackside entities
ERTMS entity	OBU, RBC or RIU.
Key archiving	Long term storage of keys, which must be authentic.
Key deletion	Deletion of keys incl. all related information and copies.
Key derivation	Derivation of a key based on another cryptographic key.
Key distribution	Confidential distribution of keys.
Key generation	Confidential generation of key related material used for encryption, decryption and key derivation.
Key installation	Confidential installation of keys into the entities.
Key management	The generation, storage, secure distribution, revocation, deletion and application of keying material.
Key registration	Registration of key related material incl. all related information.
Key storage	Storing of keys must be authentic and confidential.
Key validation	Check newly generated key for weakness or semi-weakness according to ANSI X3.92 (see ref [4]), for already existing keys and known compromised keys in the KM domain. A key which fails the validation shall be discarded and the generation shall be started again.

keying material	The data (e.g., keys) necessary to establish and maintain cryptographic keying relationships.
KMC interworking	Exchange of keys and KMS messages between KMCs.
Symmetric key	A cryptographic key which is used in symmetric keyed algorithms. A symmetric key is used for both encryption and decryption.
Secure environment	A secure place or a specially developed device. Under normal circumstances it should not be possible for unauthorised persons to read out any information from this special place or device.

6. KM CONCEPTS AND PRINCIPLES

6.1 Introduction and background

6.1.1.1 In the ERTMS system, OBU equipment and wayside RBCs or other related equipment will exchange information using the EuroRadio protocol to secure communication over an open non-trusted medium. When an ERTMS on-board equipment wishes to communicate with an RBC, it shall be able to verify that communication is established with an authorised RBC and vice versa. Consequently the authenticity and integrity of any information exchanged between ERTMS on-board equipment and RBC is also verified.

6.1.1.2 The method of ensuring that both communicating entities are the ones they assert to be, is based on an Identification and Authentication (I&A) dialogue. In order to ensure complete protection, this procedure shall take place each time the peer entities effectively start a new communication session between them.

6.1.1.3 After each successful I&A dialogue, data are protected using a Message Authentication Code (MAC). The calculation of this code is based on the existence of shared secret information only known by the entities that are actually communicating with each other.

6.1.1.4 Both I&A dialogue and MAC calculation procedures are fully specified in the Safety Functional Module described in UNISIG EuroRadio FIS (Ref [1]). These procedures are based on particular cryptographic techniques that use secret keys. However, they do not provide any means to create, distribute or update these keys. Moreover, their full efficiency relies on the key secrecy that can only be guaranteed when clear key management functions and system security policy are defined according to implementation constraints and railway operational scenarios.

6.1.1.5 This specification covers management of cryptographic keys as defined in UNISIG EuroRadio FIS (Ref [1]).

6.2 Off-line KM FIS Requirements and system requirements

6.2.1.1 The requirements for the off-line KM FIS are divided into general requirements and system requirements.

6.2.1.2 The off-line KM system requirements cover key material exchange between KMCs only. Key distribution to RBC's, OBU's and RIU's are out of the scope of this document.

6.2.2 General FIS requirements

6.2.2.1 The off-line KM FIS has to meet the following general requirements :

Number	Requirements	Type
GREQ-1	Only off-Line exchange of key related material between KM domains shall be supported.	Mandatory
GREQ-2	The KMS shall provide exchange of key related material in a secure way between KM domains.	Mandatory
GREQ-3	Theft or loss of key material shall be detected.	Mandatory
GREQ-4	The KMS shall prevent unauthorised persons from modifying keys. Such modification shall only be possible with a very high effort.	Mandatory

Table 1 - FIS requirements

6.2.3 System Requirements

6.2.3.1 The off-line KM assumes the following system requirements are met by operational KMS :

Number	Requirements	Type
SREQ-1	Key Generation: Keys for encryption and decryption or for authentication shall be generated only by authorised persons and processes in a well defined organisation and secure environment. Each generated key shall be uniquely identified. The key should be generated randomly to prevent possible prediction. The KMC must cover all these cases.	Informative, see Ref. [8]
SREQ-2	Key Validation: All keys generated by the KM domain shall be checked to guarantee that they are not weak or semi-weak according to ANSI X3.92 (see ref [4]). In all cases of key generation (first time or update), a check for existing keys or known compromised keys in the KM domain shall be performed.	Informative, see Ref. [8]
SREQ-3	Key Storage: Keys shall be stored by the KM domain in such a way that they remain authentic and confidential.	Informative, see Ref. [8]
SREQ-4	Key Exchange: The exchange of keys between KM domains shall take place only between the KMCs.	Informative, see Ref. [8]
SREQ-5	Key distribution and installation : Key distribution and installation to train or trackside entities shall be under the responsibility of the KM domain. It shall be performed in a secure way and include all related key information.	Informative, see Ref. [8]

Number	Requirements	Type
SREQ-6	Key Deletion: Key deletion shall be under the responsibility of the KM domain. It shall be performed in a secure way and include all related key information. All possible copies of the key material shall be deleted including installed keys in train or trackside entities except the key archive under KMC responsibility.	Informative, see Ref. [8]
SREQ-7	Key Archiving: All keys, key related material and associated key transactions shall be archived by the KM domain in an authentic and confidential way.	Informative, see Ref. [8]

Table 2 - KM system requirements

6.2.3.2 The fulfilment of these KM requirements by any KMC, together with compliance to this FIS, is the basis to define co-operation rules between KM domains that are able to provide and maintain proper control of all key material used in ETCS operations.

6.3 Key Hierarchy

6.3.1.1 This specification is compliant to the key hierarchy defined in UNISIG EuroRadio FIS (Ref [1]).

6.3.1.2 One further key is introduced at level 3 (the K-KMC), used for interworking between KMCs.

Level	Purpose
3 : Transport keys	KTRANS : Protection of KMS communication between KMCs and ERTMS entities K-KMC: Protection of KMS communication between KMCs
2 : Authentication keys	KMAC : Authentication of ERTMS entities (OBU and Trackside) during EuroRadio safe connection establishment.
1 : Session keys	KSMAC : Authentication of data transfer between ERTMS entities (OBU and Trackside) during a complete safe communication session.

Table 3 - Extended key hierarchy

Notes :

1. At level 3 KTRANS keys are not relevant in this FIS
2. At level 1 KSMAC keys are not relevant in this FIS

6.3.1.3 K-KMC keys are composed of two keys: K-KMC1 is used for protecting the authenticity and integrity of the messages exchanged between KMCs. K-KMC2 is used to protect by encryption the KMAC exchanged between KMCs.

6.4 General Principles

6.4.1 Key Management context

6.4.1.1 The following figure summarises the KMS context :

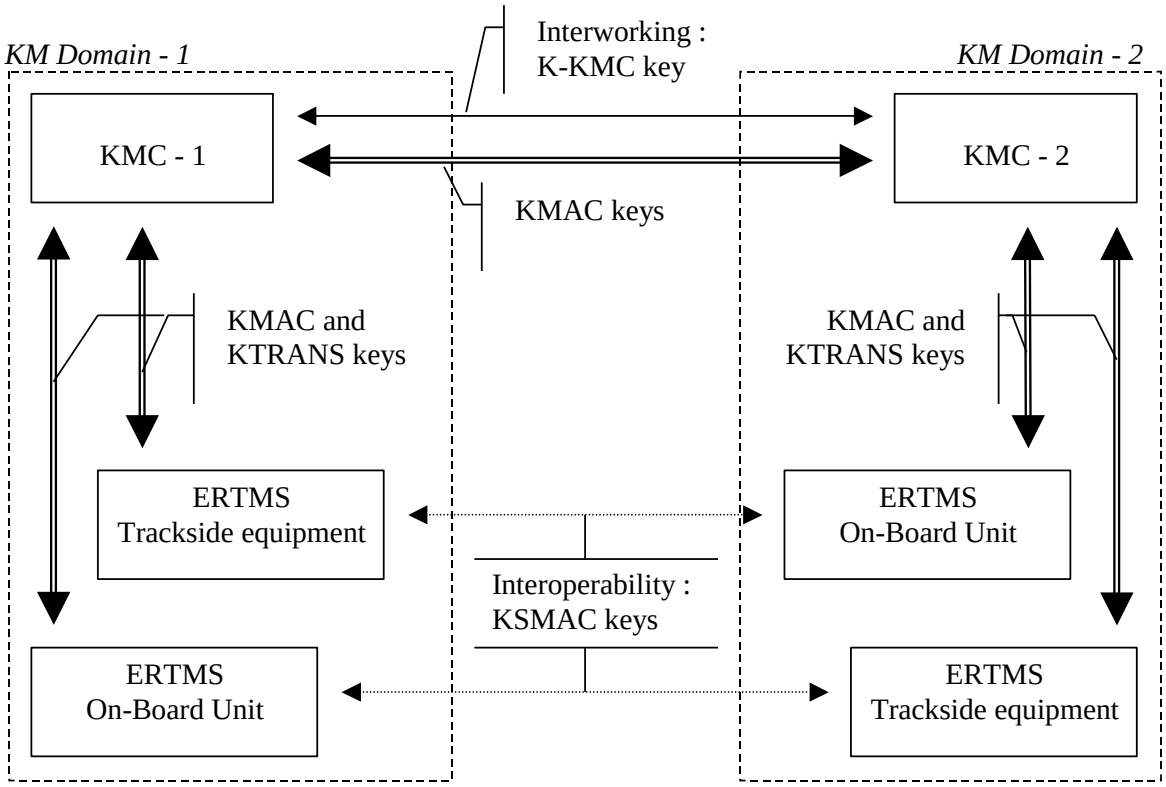

Figure 1: KM context diagram

6.4.1.2 The KM interfaces indicated in the figure are needed to store, update and delete keys (KMS interfaces).

6.4.1.3 A home KMC is responsible for distributing, updating and revoking KMAC to all trackside and on-board entities of its domain. Therefore on-board and trackside entities shall use only their home KMC for key management purposes.

6.4.1.4 A KMC is responsible for the generation of KMACs of the on-board entities which belong to its home KM domain and need to establish safe connections with trackside entities in the same domain. Depending on the adopted key assignment rule, a KMC can also be responsible for the generation of KMACs of the on-board entities which belong to foreign KM domains and need to establish safe connections with trackside entities in its home KM domain.

6.4.1.5 When an OBU entity needs to establish safe communication with a trackside entity belonging to a foreign domain, the requested keying material has to be exchanged between KMC's and distributed to the trackside and onboard entities.

6.4.1.6 It should be noted that the interface between ERTMS OBU and trackside equipment has been completely standardised for interoperability in UNISIG EuroRadio FIS (Ref [1]), where a KMAC based Identification and Authentication procedure as well as KSMAC-based communication session protection is defined.

6.4.1.7 The following table summarises the different types of key and their respective usage as referred to in the rest of this FIS:

Involved entities	Key used for Identification & Authentication	Key used for message authentication	Key used for encryption	notes
RBC – OBU	KMAC	KSMAC	---	relevant for interoperability
KMC – RBC/OBU	---	KTRANS1	KTRANS2	Domain internal
KMC – KMC	---	K-KMC1	K-KMC2	relevant for interworking

Table 4 - Usage of the defined keys

6.4.2 Procedure for CBC-MAC code calculation

6.4.2.1 The CBC-MAC is a value of 64 bits calculated on a message "m" using the K-KMC1 according to section "Generic MAC-Calculation" of UNISIG EuroRadio FIS (Ref [1]).

Note: No other variable is used for the calculation of the CBC-MAC.

6.4.2.2 The purpose of the CBC-MAC is to check the integrity and to authenticate the message "m".

6.5 Key assignment

6.5.1.1 The OBU KMAC(s) may be installed in trackside entities in any number of KM domains, including the home KM domain and any foreign KM domain in which the Onboard entity is authorised to run.

6.5.1.2 Each RBC shall be equipped with the KMACs of all the relevant OBU entities authorised to run within the RBC's area of control.

6.5.1.3 Key assignment rules are defined in Ref. [8].

7. BASIC KM FUNCTIONS

7.1 General

7.1.1.1 The following section provides the basic KM functions relevant for the proper implementation of this off-line KM FIS. Additional KM functions could exist locally but shall not impede inter-KMC key exchanges.

7.1.1.2 The Basic KM functions are mandatory.

7.2 Basic KM Functions

7.2.1.1 In order to allow interoperability between KM domains, and perform secure and consistent inter KMC key exchange, any KMS shall support the following functions:

a) Define KM domain

b) Install KMC

c) Negotiate K-KMC

d) Generate KMACs

e) Exchange KMACs with another KMC

f) Distribute KMACs

g) Update KMACs

h) Delete KMACs

i) Delete K-KMCs

j) Archive keys and KM transactions

k) Manage different types of users

7.2.2 Define KM domain

7.2.2.1 Following decisions of railways operation and approval authorities, the KMC Administrator defines which ERTMS entities shall be part of the KM domain. The KMC Administrator provides the following information :

a) ETCS-ID of the KMC

b) List of trackside entities belonging to the domain

c) List of OBU entities belonging to the domain

d) Definition of KMAC assignment rule(s)

e) Identification of other domains, where its home OBU entities should be able to be accepted

f) Identification of other domains, from which foreign OBU entities should be accepted

7.2.3 Install KMC

7.2.3.1 The KMC Administrator shall ensure the completeness of all design and provisioning tasks to set-up equipment able to perform the following operations with the intended security:

a) generate keys

b) send keys to other KMCs

c) receive keys from other KMCs

d) delete keys

e) archive keys

f) validate keys

g) produce activity log

7.2.4 Negotiate K-KMC key

7.2.4.1 The KMC Administrator shall negotiate a K-KMC for each KMC with which off-line key exchange is required.

7.2.4.2 Different solutions are possible and an agreement is needed between the involved administrations, e.g.,:

a) One KMC Administrator generates, validates the key and distributes it to the other

b) Each KMC Administrator generates a part of the key, distributes it to the other, then both validate the resulting key and use it, if the validation was positive

c) Both KMC administrators receive the key from an independent key generator, responsible for the generation and validation of the key

7.2.4.3 Whatever the adopted solution the confidentially of K-KMC shall be guaranteed outside the involved KMCs.

7.2.5 Generate KMAC

7.2.5.1 Key generation shall be performed by the KMC according to security requirements of interoperable ERTMS applications. Specific technical solutions do not need harmonisation (provided security is ensured).

7.2.5.2 The generated KMACs for interoperable entities shall be validated by the generating KMC.

7.2.5.3 A unique serial number shall be associated with each generated KMAC.

7.2.6 Exchange KMAC with another KMC

7.2.6.1 This function shall be used by the issuing KMC only to install a key in the receiving KMC.

7.2.6.2 The issuing KMC Administrator decides when it is appropriate to exchange the KMAC. This decision is taken according to a predefined key renewal plan or in case of detection of hazardous situations (loss of confidentiality).

7.2.6.3 The key renewal plan shall be agreed in advance as part of the inter KMC agreement.

7.2.6.4 Issuing KMC:

7.2.6.4.1 The issuing KMC Administrator shall determine which OBU entity that KMAC is intended for and which RBC entities will be accepting the OBU.

7.2.6.5 Receiving KMC:

7.2.6.5.1 The receiving KMC Administrator shall confirm that the request has been received and is consistent, and shall take the necessary actions to put the KMACs in operation.

7.2.6.6 The key shall be distributed confidentially by encryption with K-KMC2.

7.2.7 Update KMAC

7.2.7.1 The KMC Administrator shall be able to update the validity period and the list of ETCS entities of an existing (already issued) KMAC.

7.2.7.2 The list of ETCS entities and the validity period included in the update request shall replace the previous ones.

7.2.7.3 Only the KMC that issued the KMAC exchange request shall be able to update key related information.

7.2.7.4 It should be possible to update KMACs during both maintenance and normal operation.

Note 1: updating during maintenance is intended for introduction or removal of ERTMS entities.

Note 2: updating during maintenance is an implementation issue; standardisation for interoperability is not needed.

7.2.7.5 The receiving KMC Administrator shall confirm to the originator of the request that the request has been received and is consistent, and shall take the necessary actions to update the KMACs.

7.2.8 Delete KMAC

7.2.8.1 The KMC Administrator shall be able to request other KMC Administrator(s) to delete keys. The reason of the deletion request shall be indicated.

7.2.8.2 In case of key deletion, the KMC Administrator shall ensure that all copies of the key, including all related information, are deleted (operational issue).

7.2.8.3 The receiving KMC Administrator shall confirm to the originator of the key deletion that the request or notification has been received and is consistent.

7.2.8.4 The KMAC deletion could be triggered either by the KMAC issuing KMC using a deletion request, or by the KMAC receiving KMC using a deletion notification. See ref. [8].

7.2.9 Archive keys and KM transactions

7.2.9.1 The KMC Administrator shall store confidentially all information on the produced keys, including:

a) Assignment of keys to entities

b) State of the key (e.g. currently used, deleted, compromised, waiting exchange/deletion confirmation)

7.2.10 Delete K-KMC

7.2.10.1 The KMC Administrator shall inform other KMC Administrator(s) about deleted K-KMC keys.

7.2.10.2 The KMC Administrator shall ensure that all copies of the key, including all related information, are deleted (security and operational issue).

7.2.11 Manage several types of users

7.2.11.1 The KMC shall be able to manage following types of user as specified in Ref. [8] :

- KMC Administrator;
- KMC Operator;
- KMC Maintainer.

8. OFF-LINE KM TRANSACTIONS

8.1 General

8.1.1.1 The off-line KM transactions have to support the exchange, update and deletion of KMACs between interoperable KM domains.

8.1.1.2 A KMC and its associated KM domain shall be identified using a unique ETCS ID expanded according to rules defined in UNISIG EuroRadio FIS (Ref [1] §8.2.4.6).

8.1.1.3 The ETCS ID expanded is composed of ETCS ID type and ETCS ID. In the next sections, the ETCS ID expanded shall be coded as follows :

7654 3210 (bit)	Encoding Fields
xxxx xxxx	ETCS ID type (1 byte)
yyyy yyyy yyyy yyyy yyyy yyyy	ETCS ID (3 bytes)

Table 5 – Encoding of ETCS ID expanded

8.1.1.4 The following KM transactions are specified according to KM requirements established in previous paragraphs.

8.1.1.5 It is assumed that the K-KMC has been previously exchanged securely between KMCs.

8.2 Off-line KM messages definition

8.2.1.1 The off-line Key Management System shall support the following KM messages:

- KMAC exchange
- KMAC deletion
- KMAC update
- KMAC exchange confirmation
- KMAC deletion confirmation
- KMAC update confirmation
- KMAC negative acknowledgment

8.2.1.2 The following table defines the codification of the KM messages :

Number	MESSAGE TYPE	MESSAGE CODE 7654 3210 (bit)	Comment
1	KMAC-EXCHANGE	0000 0100	KMAC Exchange request between off-line KM domains.
2	KMAC-DELETION	0000 0110	KMAC deletion between off-line KM domains. This service is used for KMAC deletion request and notification.
3	KMAC-UPDATE	0001 0000	KMAC parameters (validity period and list of ETCS entities) update between off-line KM domains.
4	CONF-KMAC-EXCHANGE	0000 0101	The off-line KM domains confirm the reception and consistency of the KMAC exchange request.
5	CONF-KMAC-DELETION	0000 0111	The off-line KM domains confirm : • the reception and consistency of the KMAC deletion transaction, or • the reception and consistency of the KMAC deletion notification.
6	CONF-KMAC-UPDATE	0001 0001	The off-line KM domains confirm the reception and consistency of the KMAC update request.
7	KMAC-NEGACK	0000 0000	The off-line KM domains reject the KM message by sending a negative acknowledgment

Table 6 – Off-line KM messages

8.3 Encryption, integrity and authentication of KM services

8.3.1 Key definition

8.3.1.1 In this document a triple-key is defined as an array of 192 bits consisting of 3 concatenated DES-keys K1, K2, K3 of 64-bit length each. In short: triple-key = K1 | K2 | K3, where the symbol "|" means concatenation.

8.3.1.2 Bit 0 is the left most bit of the triple-key.

8.3.1.3 For a valid triple-key, each eighth bit (LSB = bit 7, bit 15, …, bit 191) of the 192-bits must be set to an odd-parity value as defined in the standard ref. [4].

8.3.1.4 The following table shows the structure of a triple-key:

triple-Key length 192 bit		
$b_0, b_1,$ … b_{191}		
DES-key K1 length 64 bit	DES-key K2 length 64 bit	DES-key K3 length 64 bit
$b_0, \ldots b_{63}$	$b_{64}, \ldots b_{127}$	$b_{128}, \ldots b_{191}$

Table 7 Structure of triple key

8.3.2 Structure of K-KMC, KMAC and encrypted KMAC

8.3.2.1 The K-KMC shall be negotiated between KM domains before any interoperable KM service is started.

8.3.2.2 The purpose of K-KMC is to protect the KM service transactions providing encryption, integrity check and authentication of exchanged KMACs.

8.3.2.3 The K-KMC is defined as an array of 384 bits consisting of two concatenated triple-keys named K-KMC1 and K-KMC2. In short: K-KMC = K-KMC1 | K-KMC2.

8.3.2.4 The first part of the K-KMC is K-KMC1 (192 Bit length). This KMC1 is used to check integrity and authenticate the KM service transaction using CBC-MAC. The procedure for MAC calculation is described in § 6.4.2.

8.3.2.4.1 The second part of the K-KMC is K-KMC2 (192 Bit length). This K-KMC2 is used to encrypt the KMAC to ensure confidentiality during KM service message exchange. The procedure for KMAC encryption is described here below.

8.3.2.5 The KMAC is a triple-key composed of 3 DES-keys : K1 | K2 | K3.

8.3.2.6 Considering this KMAC, the encrypted KMAC is a triple-key composed of the following keys : Encrypted(K1)| Encrypted(K2)| Encrypted(K3) .

8.3.3 Procedure for KMAC encryption and decryption

8.3.3.1 Each DES-key of the KMAC is encrypted and decrypted using the K-KMC2 according to the Triple-DES process :

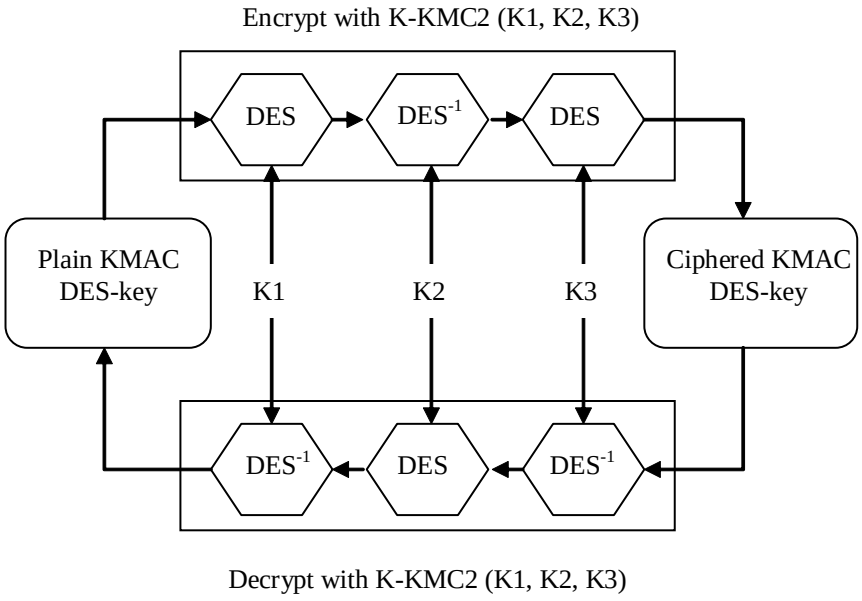

Figure 2 : KMAC encryption using K-KMC2 K1, K2 ,K3 and Triple-DES

8.3.3.2 The Triple-DES algorithm is used in mode ECB. Encryption/Decryption corresponds to three DES operations in the sequence encrypt-decrypt-encrypt for encryption and decrypt-encrypt-decrypt for decryption and using keying option 1 , i.e. three different keys K1, K2, K3. (see also Ref [3] § 6.1 & 6.2).

8.3.3.3 The DES refers to Data Encryption Algorithm specified in Ref [4].

Note : An example of a DES key encryption is available in section “Predefined key” of KMC-ETCS Entity Offline KM FIS (Ref [2]).

8.4 KMAC transaction message format

8.4.1 Introduction

8.4.1.1 This section describes the detailed format of the KMAC transaction messages.

8.4.1.2 Note 1: all ETCS-IDs used in the message format are ETCS IDs expanded and are encoded as per Table 5.

8.4.1.3 Note 2: the field “OB-ETCS-ID” can be replaced by an RBC ETCS-Id in the case that KMAC transaction messages are used to manage keys related to the RBC-RBC interface. See ref. [9].

8.4.1.4 Note 3: all date values used in the transaction messages shall refer by default to UTC.

8.4.2 KMAC exchange transaction format

8.4.2.1 The structure and contents of the KMAC exchange transaction are described in the next table. The interoperable KMACs must be exchanged in an authentic and confidential way. The following actions for KMAC preparation are needed before the KMAC can be exchanged by the issuer off-line Key Management Centre:

1. MESSAGETYPE — Set as KMAC exchange message type (see KM message Definition),
2. OB-ETCS-ID — Read ETCS-ID expanded in KM database for the on-board entity and store in structure,
3. TR-QUANT — Establish number k of concerned trackside entity and store in structure,
4. TR-ETCS-IDi — Read ETCS-ID expanded in KM database for each concerned trackside entity (from i = 1 to k) and store in structure,
5. KM ETCS-ID1 — Read own KM ETCS-ID expanded and store in structure,
6. KM ETCS-ID2 — Read foreign KM ETCS-ID expanded from KM database and store in structure,
7. ISSUE-DATE — Use current date and store in structure,
8. VALID-PERIOD[1] — Use start and end date defined for interoperable service and store in structure,
9. TNUM — Increment KMAC exchange transaction number and store in structure,
10. KMAC — Generate and validate KMAC related to interoperable service (optional),

    Note 1: the parity bits of KMAC must be set according to ANSI X3.92 (see ref [4]).

    Note 2: this action is not required if an existing KMAC is used.
11. ENC(KMAC) — Encrypt KMAC using K-KMC2 and store result in structure,
12. SNUM — Increment KMAC serial number and store in structure
13. CBC-MAC — Calculate CBC-MAC using K-KMC1 and store result in structure,

8.4.2.2 These actions shall be performed for each single KMAC exchange between KM interoperable domains.

8.4.2.3 The following table describes the format of KMAC exchange request:

[1] The end date of interoperable service shall be understood “at the latest”. Key updates shall be planned and implemented before the end date in order to avoid any perturbation to railway operation.

Octet	Bit 7654 3210	Field name	Field
1	0000 0100	KMAC-EXCHANGE	KMAC Exchange request.
2 3 4 5	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	OB-ETCS-ID	The ETCS-ID expanded of the On-Board ERTMS entity (e.g. an OBU ETCS-ID).
6	xxxx xxxx	TR-QUANT	The number k of Trackside ERTMS Identities concerned by the interoperable service and listed in the next field (0000 0000 indicates an empty list)
3 + 4*i 4 + 4*i 5 + 4*i 6 + 4*i	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	TR-ETCS-IDi	The ETCS-ID expanded of the Trackside ERTMS entity (i) (e.g. a RBC ETCS-ID) i = 1 to k Note : If k is equal to “0”, this field is not present.
7 + 4*k 8 + 4*k 9 + 4*k 10 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID1	The ETCS-ID expanded of the issuer KMC ERTMS entity of the KMAC exchange request
11 + 4*k 12 + 4*k 13 + 4*k 14 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID2	The ETCS-ID expanded of the destination KMC ERTMS entity of the KMAC exchange request
15 + 4*k 16 + 4*k 17 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	ISSUE-DATE	The date of production of the KM exchange request (in DD MM YY format e.g. 01 01 05 and BCD coded)
18 + 4*k 19 + 4*k 20 + 4*k 21 + 4*k 22 + 4*k 23 + 4*k 24 + 4*k 25 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	VALID-PERIOD	Beginning followed by end dates of validity period for the KMAC (in HH DD MM YY format e.g. 03 01 01 05 which means 1st January 2005 at 3:00 AM and BCD coded) REM : FF FF FF FF means infinite end date
26 + 4*k	xxxx xxxx	TNUM	This Transaction Number identifies unambiguously the KMAC exchange transaction. This shall be used in the confirmation to ensure verification by the KMC issuing the exchange. 0000 0000 value is not used- All other values can be used.
27 + 4*k ... 50 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	ENC(KMAC)	The KMAC of the entity is stored encrypted (see encryption procedure using K-KMC2) by the issuer off-line Key Management Centre.
51 + 4*k 52 + 4*k 53 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	SNUM	This Serial Number together with the issuer KMC Id identifies unambiguously the exchanged KMAC (MSB first)
54 + 4*k 55 + 4*k 56 + 4*k 57 + 4*k 58 + 4*k 59 + 4*k 60 + 4*k 61 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	CBC-MAC	The CBC-MAC shall be calculated over the message from octet 1 up to but excluding the CBC-MAC field. The CBC-MAC shall be calculated using K-KMC1 and according to § 6.4.2.

Table 8 Structure Information of KMAC exchange request

8.4.2.4 After receiving the KMAC exchange request, the destination KMC shall perform the following actions regarding the received KMAC in order to prepare the KMAC exchange confirmation:

1. MESSAGETYPE — Set as KMAC exchange confirmation message (see KM message Definition),
2. CBC-MAC — Select K-KMC according issuer KMC identity of KMAC exchange transaction and calculate CBC-MAC using K-KMC1, If the result of CBC-MAC calculation is valid (identical to received CBC-MAC), then continue, else apply local error recovery and send back KMAC negative acknowledgment,
3. KM-ETCS-ID1 — Read ETCS-ID expanded of the destination KMC entity of KMAC exchange transaction, Verify that the destination KMC identity is correct (= the receiver KMC) and store as issuer KMC identity in structure of confirmation,
4. KM-ETCS-ID2 — Read ETCS-ID expanded of the issuer KMC entity of KMAC exchange transaction and store as destination KMC identity in structure of confirmation,
5. OB-ETCS-ID — Verify the ETCS-ID expanded of the On-Board Unit is in receiver KMC database and store in structure of confirmation, else apply local error recovery and send back KMAC negative acknowledgment,

   Note : the OB-ETCS-ID should have been introduced already in KM database after railway administrative arrangements.
6. TR-QUANT — Use this number to read eventual list of trackside ERTMS entities and store in structure of confirmation,
7. TR-ETCS-IDi — Read ETCS-ID expanded for each trackside entity (i = 1 to TR-QUANT), store in KM database and store in structure of confirmation,
8. ISSUE-DATE — Archive KMAC exchange transaction adding current date in KM database, store current date as new ISSUE-DATE in structure of confirmation
9. VALID-PERIOD — Check validity start and end date, in case of incoherent date the received KMAC shall be rejected by the receiver KMC.
10. TNUM — Use the TNUM of the KM exchange transaction and store in structure of confirmation,
11. DEC(KMAC) — Decrypt received KMAC using K-KMC2 and store result in KMC database.

Note 1: KMC database is considered protected against security attack.

Note 2: the parity bits of KMAC should be verified according to ANSI X3.92 (see ref [4]) If the result of the verification is valid, then continue, else apply local error recovery and send back KMAC negative acknowledgment.

12. CBC-MAC Calculate CBC-MAC using K-KMC1 and store result in structure of confirmation,

8.4.2.5 These actions shall be performed for each single KMAC exchange confirmation between KM interoperable domains.

8.4.2.6 The following table describes the format of KMAC exchange confirmation :

Octet	Bit 7654 3210	Field name	Field
1	0000 0101	CONF-KMAC-EXCHANGE	KMAC Exchange confirmation message.
2 3 4 5	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	OB-ETCS-ID	The ETCS-ID expanded of the On-Board ERTMS entity (e.g. an OBU ETCS-ID).
6	xxxx xxxx	TR-QUANT	The number k of Trackside ERTMS Identities concerned by the interoperable service and listed in the next field (0000 0000 indicates an empty list)
3 + 4*i 4 + 4*i 5 + 4*i 6 + 4*i	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	TR-ETCS-IDi	The ETCS-ID expanded of the Trackside ERTMS entity (i) (e.g. a RBC ETCS-ID) i = 1 to k Notes : The list of ETCS-ID expanded shall be equal to the one sent in the exchange request; If k is equal to “0”, this field is not present
7 + 4*k 8 + 4*k 9 + 4*k 10 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID1	The ETCS-ID expanded of the issuer KMC ERTMS entity of the KMAC exchange confirmation
11 + 4*k 12 + 4*k 13 + 4*k 14 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID2	The ETCS-ID expanded of the destination KMC ERTMS entity of the KMAC exchange confirmation
15 + 4*k 16 + 4*k 17 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	ISSUE-DATE	The date of production of the KM exchange confirmation (in DD MM YY format e.g. 01 01 05 and BCD coded)
18 + 4*k	xxxx xxxx	TNUM	This Transaction Number identifies unambiguously the confirmation and is equal to the TNUM received in the KM exchange transaction.
19 + 4*k 20 + 4*k 21 + 4*k 22 + 4*k 23 + 4*k 24 + 4*k 25 + 4*k 26 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	CBC-MAC	The CBC-MAC shall be calculated over the message from octet 1 up to but excluding the CBC-MAC field. The CBC-MAC shall be calculated using K-KMC1 and according to § 6.4.2.

Table 9 Structure Information of KMAC exchange confirmation

NOTE: the KMAC exchange confirmation shall be sent only after having received and checked the consistency of the KMAC exchange request. The conditions regarding KMAC installation (e.g. performance and timing) are out of scope of this FIS and should be defined in the KMC agreement between KMCs.

8.4.3 KMAC deletion transaction format

8.4.3.1 The structure and contents of the KMAC deletion transaction are described in the next table. The following actions for KMAC deletion are needed before the KMAC can be deleted by the issuing off-line Key Management Centre:

1. MESSAGETYPE Set as KMAC deletion message (see KM message Definition),
2. SUBTYPE Precise deletion request or notification and store in structure,
3. OB-ETCS-ID Read in KM database the ETCS-ID expanded of the on-board entity and store in structure ,
4. TR-QUANT Establish number k of trackside entity to which the key has been distributed and store in structure,
5. TR-ETCS-IDi Read in KM database ETCS-ID expanded of each concerned trackside entity (i = 1 to k) and store in structure,
6. KM ETCS-ID1 Read own KM ETCS-ID expanded and store in structure,
7. KM ETCS-ID2 Read foreign KM ETCS-ID expanded from KM database and store in structure,
8. ISSUE-DATE Use current date and store in structure,
9. EFF-DATE Use start date defined for key deletion service and store in structure,

   Note : in case of deletion notification this date can be equal or precede ISSUE-DATE
10. TNUM Increment KMAC deletion transaction number and store in structure,
11. SNUM Use deleted KMAC serial number and store in structure,
12. REASON Define reason for KMAC deletion and store in structure,
13. CBC-MAC Calculate CBC-MAC using K-KMC1 and store result in structure,

8.4.3.2 These actions shall be performed for each single KMAC deletion request or notification between KM interoperable domains.

8.4.3.3 The following table describes the format of KMAC deletion request or notification :

Octet	Bit 7654 3210	Field name	Field
1	0000 0110	KMAC-DELETION	KMAC Deletion message

Octet	Bit 7654 3210	Field name	Field
2	xxxx xxxx	SUBTYPE	0000 0010 Deletion request 0000 0100 Deletion notification Note : Other values not permitted
3 4 5 6	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	OB-ETCS-ID	The ETCS-ID expanded of the On-Board ERTMS entity (e.g. an OBU ETCS-ID).
7	xxxx xxxx	TR-QUANT	The number k of Trackside ERTMS Identities concerned by the interoperable service and listed in the next field (0000 0000 indicates an empty list)
4 + 4*i 5 + 4*i 6 + 4*i 7 + 4*i	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	TR-ETCS-IDi	The ETCS-ID expanded of the Trackside ERTMS entity (i) (e.g. a RBC ETCS-ID) i = 1 to k Note : If k is equal to “0”, this field is not present.
8 + 4*k 9 + 4*k 10 + 4*k 11 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID1	The ETCS-ID expanded of the issuer KMC ERTMS entity of the KMAC deletion request or notification
12 + 4*k 13 + 4*k 14 + 4*k 15 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID2	The ETCS-ID expanded of the destination KMC ERTMS entity of the KMAC deletion request or notification
16 + 4*k 17 + 4*k 18 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	ISSUE-DATE	The date of production of the KM deletion request or notification (in DD MM YY format e.g. 01 01 05 and BCD coded)
19 + 4*k 20 + 4*k 21 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	EFF-DATE	The date by which the KMAC deletion is performed (in DD MM YY format e.g. 01 01 05 and BCD coded)
22 + 4*k	xxxx xxxx	TNUM	This Transaction Number identifies unambiguously the KMAC deletion transaction. This shall be used in the confirmation to ensure verification by the KMC issuing the deletion request or notification. 0000 0000 value is not used- All other values can be used.
23 + 4*k 24 + 4*k 25 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	SNUM	This Serial Number together with the issuer KMC Id in case of deletion request, or together with the destination KMC ETCS ID in case of deletion confirmation, identifies unambiguously the KMAC to be deleted (MSB first).
26 + 4*k	xxxx xxxx	REASON	Prescribed values are : 0000 0001 Termination of KM interoperable service 0000 0010 KMAC compromised Note : other values are reserved for bilateral agreement in KMC agreement
27 + 4*K 28 + 4*k 29 + 4*k 30 + 4*k 31 +4*k 32 + 4*k 33 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	CBC-MAC	The CBC-MAC shall be calculated over the message from octet 1 up to but excluding the CBC-MAC field. The CBC-MAC shall be calculated using K-KMC1 and according to § 6.4.2.

Octet	Bit 7654 3210	Field name	Field
34 + 4*k	xxxx xxxx		

Table 10 Structure Information of KMAC deletion

NOTE: In case of deletion notification, the KMAC deletion notification shall be sent after the deletion of all installed copies of KMAC inside the notifying KM domain is completed. The conditions regarding KMAC deletion (e.g. performance and timing) must be defined in the KMC agreement.

8.4.3.4 After receiving the KMAC deletion information the destination KMC shall perform the following actions regarding the deletion in order to prepare the KMAC deletion confirmation:

1. MESSAGETYPE Set as KMAC deletion confirmation message (see KM message Definition),
2. CBC-MAC Select K-KMC according issuer KMC identity of KMAC deletion transaction and calculate CBC-MAC using K-KMC1, If the result of CBC-MAC calculation is valid (identical to received CBC-MAC), then continue, else apply local error recovery and send back KMAC negative acknowledgment,
3. KM-ETCS-ID1 Read the ETCS-ID expanded of the destination KMC entity of KMAC deletion transaction, Verify that the destination KMC identity is correct (= the receiver KMC) and store as issuer KMC identity in structure of confirmation,
4. KM-ETCS-ID2 Read the ETCS-ID expanded of the issuer KMC entity of KMAC deletion transaction and store as destination KMC identity in structure of confirmation,
5. SUBTYPE Read the SUBTYPE of the KM deletion transaction and store in structure of the confirmation
6. OB-ETCS-ID Verify the ETCS-ID expanded of the OB-ETCS-ID is in receiver KM database and store in structure of confirmation, else apply local error recovery and send back KMAC negative acknowledgment,
7. TR-QUANT Use this number to read eventual list of trackside ERTMS entities and store in structure of confirmation,
8. TR-ETCS-IDi Read the ETCS-ID expanded for each trackside entity (i = 1 to TR-QUANT), store in KM database and store in structure of confirmation,
9. ISSUE-DATE Archive KMAC deletion transaction adding current date in KM database, store current date as new ISSUE-DATE in structure of confirmation,
10. TNUM Use the TNUM of the KM deletion transaction and store in structure of confirmation,

11. SNUM — Read the SNUM field and use it together with the issuer KMC Id in case of deletion request, or together with the destination KMC ETCS ID in case of deletion confirmation, to retrieve the relevant KMAC to be deleted.

12. CBC-MAC — Calculate CBC-MAC using K-KMC1 and store result in structure of confirmation,

8.4.3.5 These actions shall be performed for each single KMAC deletion request or notification confirmation between KM interoperable domains.

8.4.3.6 The following table describes the format of KMAC deletion request or notification confirmation :

Octet	Bit 7654 3210	Field name	Field
1	0000 0111	CONF-KMAC-DELETION	KMAC Deletion confirmation message
2	xxxx xxxx	SUBTYPE	0000 0010 Deletion request 0000 0100 Deletion notification Note : Other values not permitted
3 4 5 6	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	OB-ETCS-ID	The ETCS-ID expanded of the On-Board ERTMS entity (e.g. an OBU ETCS-ID).
7	xxxx xxxx	TR-QUANT	The number k of Trackside ERTMS Identities concerned by the interoperable service and listed in the next field (0000 0000 indicates an empty list)
4 + 4*i 5 + 4*i 6 + 4*i 7 + 4*i	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	TR-ETCS-Idi	The ETCS-ID expanded of the Trackside ERTMS entity (i) (e.g. a RBC ETCS-ID) i = 1 to k Note : If k is equal to “0”, this field is not present.
8 + 4*k 9 + 4*k 10 + 4*k 11 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID1	The ETCS-ID expanded of the issuer KMC ERTMS entity of the KMAC deletion request of notification confirmation Note : The list of ETCS-ID expanded shall be equal to the one sent in the deletion request or notification.
12 + 4*k 13 + 4*k 14 + 4*k 15 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID2	The ETCS-ID expanded of the destination KMC ERTMS entity of the KMAC exchange confirmation deletion request of notification confirmation
16 + 4*k 17 + 4*k 18 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	ISSUE-DATE	The date of production of the KM deletion request or notification confirmation (in DD MM YY format e.g. 01 01 05 and BCD coded)
19 + 4*k	xxxx xxxx	TNUM	This Transaction Number identifies unambiguously the confirmation and is equal to the TNUM received in the KM deletion transaction.
20 + 4*k 21 + 4*k 22 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	CBC-MAC	The CBC-MAC shall be calculated over the message from octet 1 up to but excluding the CBC-MAC field. The CBC-MAC shall be calculated using K-KMC1 and according to § 6.4.2.

Octet	Bit 7654 3210	Field name	Field
23 + 4*k 24 + 4*k 25 + 4*k 26 + 4*k 27 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx		

Table 11 Structure Information of KMAC deletion confirmation

NOTE: The KMAC deletion confirmation shall be sent only after having received and checked the consistency of the KMAC deletion request or notification. The conditions regarding KMAC deletion (e.g. performance and timing) must be defined in the KMC agreement.

8.4.4 KMAC Update transaction format

8.4.4.1 The structure and contents of the KMAC update transaction are described in the next table. The interoperable KMACs must be updated in an authentic and confidential way. The following actions for KMAC preparation are needed before KMAC update can be done by the issuer off-line Key Management Centre:

1. MESSAGETYPE Set as KMAC update message type (see KM message Definition),
2. OB-ETCS-ID Read in KM database the ETCS-ID expanded of the on-board entity and store in structure,
3. TR-QUANT Establish number k of concerned trackside entity and store in structure,
   Note: a value of 0 indicates an empty list. This means that the relevant KMAC won't be used with any ETCS-ID. The KMAC, however, is still present in the KM database. So an Update function with 0 TR_QUANT is NOT equivalent to a Delete function.
4. TR-ETCS-IDi Read in KM database the ETCS-ID expanded for each concerned trackside entity (from i = 1 to k) and store in structure,
5. KM ETCS-ID1 Read the ETCS-ID expanded of own KM ETCS-ID and store in structure,
6. KM ETCS-ID2 Read the ETCS-ID expanded of foreign KM ETCS-ID from KM database and store in structure,
7. ISSUE-DATE Use current date and store in structure,
8. VALID-PERIOD[2] Use start and end date defined for interoperable service and store in structure,

[2] The end date of interoperable service shall be understood "at the latest". Key updates shall be planned and implemented before the end date in order to avoid any perturbation to railway operation.

9. TNUM — Increment KMAC update transaction number and store in structure,
10. ENC(KMAC) — Encrypt KMAC using K-KMC2 and store result in structure,
11. SNUM — Read in KM database the SNUM of the affected KMAC and store in structure
12. REASON — Define reason for KMAC update and store in structure,
13. CBC-MAC — Calculate CBC-MAC using K-KMC1 and store result in structure,

8.4.4.2 These actions shall be performed for each single KMAC update between KM interoperable domains.

8.4.4.3 The following table describes the format of KMAC update request :

Octet	Bit 7654 3210	Field name	Field
1	0001 0000	KMAC-UPDATE	KMAC Update request.
2 3 4 5	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	OB-ETCS-ID	The ETCS-ID expanded of the On-Board ERTMS entity (e.g. an OBU ETCS-ID).
6	xxxx xxxx	TR-QUANT	The number k of Trackside ERTMS Identities concerned by the interoperable service and listed in the next field (0000 0000 indicates an empty list)
3 + 4*i 4 + 4*i 5 + 4*i 6 + 4*i	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	TR-ETCS-IDi	The ETCS-ID expanded of the Trackside ERTMS entity (i) (e.g. a RBC ETCS-ID) i = 1 to k Note : If k is equal to “0”, this field is not present.
7 + 4*k 8 + 4*k 9 + 4*k 10 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID1	The ETCS-ID expanded of the issuer KMC ERTMS entity of the KMAC update request
11 + 4*k 12 + 4*k 13 + 4*k 14 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID2	The ETCS-ID expanded of the destination KMC ERTMS entity of the KMAC update request
15 + 4*k 16 + 4*k 17 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	ISSUE-DATE	The date of production of the KM update request (in DD MM YY format e.g. 01 01 05 and BCD coded)
18+ 4*k 19 + 4*k 20 + 4*k 21 + 4*k 22 + 4*k 23 + 4*k 24 + 4*k 25 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	VALID-PERIOD	Beginning followed by end dates of validity period for the KMAC (in HH DD MM YY format e.g. 03 01 01 05 which means 1st January 2005 at 3:00 AM and BCD coded) REM : FF FF FF FF means infinite end date

Octet	Bit 7654 3210	Field name	Field
26 + 4*k	xxxx xxxx	TNUM	This Transaction Number identifies unambiguously the KMAC update transaction. This shall be used in the confirmation to ensure verification by the KMC issuing the update request. 0000 0000 value is not used- All other values can be used.
27 + 4*k ... 50 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	ENC(KMAC)	The KMAC of the entity is stored encrypted (see encryption procedure using K-KMC2) by the issuer off-line Key Management Centre.
51 + 4*k 52 + 4*k 53 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	SNUM	This Serial Number together with the issuer KMC Id identifies unambiguously the exchanged KMAC (MSB first)
54 + 4*k	xxxx xxxx	REASON	Prescribed values are : 0000 0001 Validity period changed 0000 0010 No more used 0000 0011 List of entities changed 0000 0100 Both validity period & list of entities changed Note : other values are reserved for bilateral agreement in KMC agreement
55 + 4*k 56 + 4*k 57 + 4*k 58 + 4*k 59 + 4*k 60 + 4*k 61 + 4*k 62 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	CBC-MAC	The CBC-MAC shall be calculated over the message from octet 1 up to but excluding the CBC-MAC field. The CBC-MAC shall be calculated using K-KMC1 and according to § 6.4.2.

Table 12 Structure Information of KMAC update request

8.4.4.4 After receiving the KMAC update request the destination KMC shall perform the following actions regarding the received KMAC in order to prepare the KMAC update confirmation:

1. MESSAGETYPE Set as KMAC update confirmation message (see KM message Definition),
2. CBC-MAC Select K-KMC according issuer KMC identity of KMAC update transaction and calculate CBC-MAC using K-KMC1, If the result of CBC-MAC calculation is valid (identical to received CBC-MAC), then continue, else apply local error recovery and send back KMAC negative acknowledgment,
3. KM-ETCS-ID1 Read the ETCS-ID expanded of destination KMC entity of KMAC update transaction, Verify that the destination KMC identity is correct (= the receiver KMC) and store as issuer KMC identity in structure of confirmation,
4. KM-ETCS-ID2 Read the ETCS-ID expanded of issuer KMC entity of KMAC update transaction and store as destination KMC identity in structure of confirmation,

5. OB-ETCS-ID — Verify the ETCS-ID expanded of On-Board Unit is in receiver KMC database and store in structure of confirmation, else apply local error recovery and send back KMAC negative acknowledgment. Note : the OB-ETCS-ID should have been introduced already in KM database after railway administrative arrangements,
6. TR-QUANT — Use this number to read eventual list of trackside ERTMS entities and store in structure of confirmation,
7. TR-ETCS-IDi — Read the ETCS-ID expanded for each trackside entity (i = 1 to TR-QUANT), store in KM database and store in structure of confirmation,
8. ISSUE-DATE — Archive KMAC exchange transaction adding current date in KM database, store current date as new ISSUE-DATE in structure of confirmation
9. VALID-PERIOD — Check validity start and end date, in case of incoherent date the received KMAC shall be rejected by the receiver KMC.
10. TNUM — Use the TNUM of the KM exchange transaction and store in structure of confirmation,
11. SNUM — Read the SNUM field, and use it together with the issuer KMC Id to retrieve the relevant KMAC.
12. CBC-MAC — Calculate CBC-MAC using K-KMC1 and store result in structure of confirmation,

8.4.4.5 These actions shall be performed for each single KMAC update confirmation between KM interoperable domains.

8.4.4.6 The following table describes the format of KMAC update confirmation :

Octet	Bit 7654 3210	Field name	Field
1	0001 0001	CONF-KMAC-UPDATE	KMAC Update confirmation message.
2 3 4 5	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	OB-ETCS-ID	The ETCS-ID expanded of the On-Board ERTMS entity (e.g. an OBU ETCS-ID).
6	xxxx xxxx	TR-QUANT	The number k of Trackside ERTMS Identities concerned by the interoperable service and listed in the next field (0000 0000 indicates an empty list)
3 + 4*i 4 + 4*i 5 + 4*i 6 + 4*i	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	TR-ETCS-IDi	The ETCS-ID expanded of the Trackside ERTMS entity (i) (e.g. a RBC ETCS-ID) i = 1 to k Notes : The list of ETCS-ID expanded shall be equal to the one sent in the update request. If k is equal to “0”, this field is not present.
7 + 4*k	xxxx xxxx	KM-ETCS-ID1	The ETCS-ID expanded of the issuer KMC ERTMS entity of the KMAC update confirmation

Octet	Bit 7654 3210	Field name	Field
8 + 4*k 9 + 4*k 10 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx		
11 + 4*k 12 + 4*k 13 + 4*k 14 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID2	The ETCS-ID expanded of the destination KMC ERTMS entity of the KMAC update confirmation
15 + 4*k 16 + 4*k 17 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx	ISSUE-DATE	The date of production of the KM update confirmation (in DD MM YY format e.g. 01 01 05 and BCD coded)
18 + 4*k	xxxx xxxx	TNUM	This Transaction Number identifies unambiguously the confirmation and is equal to the TNUM received in the KM update request.
19 + 4*k 20 + 4*k 21 + 4*k 22 + 4*k 23 + 4*k 24 + 4*k 25 + 4*k 26 + 4*k	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	CBC-MAC	The CBC-MAC shall be calculated over the message from octet 1 up to but excluding the CBC-MAC field. The CBC-MAC shall be calculated using K-KMC1 and according to §6.4.2.

Table 13 Structure Information of KMAC update confirmation

NOTE: the KMAC update confirmation shall be sent only after having received and checked the consistency of the KMAC update request. The conditions regarding KMAC installation (e.g. performance and timing) are out of scope of this FIS and should be defined in the KMC agreement between KMCs.

8.4.5 KMAC negative acknowledgment format

8.4.5.1 The structure and contents of the KMAC negative acknowledgment are described in the next table. The interoperable KMAC transaction is aborted by the destination KMC which performs the following actions to prepare the KMAC negative acknowledgment :

1. MESSAGETYPE Set as KMAC negative acknowledgment type (see KM message Definition),
2. AB-MESSAGE Read MESSAGETYPE of KMAC aborted message and store in structure,
3. OB-ETCS-ID Read in aborted KM message the ETCS-ID expanded of the on-board entity and store in structure,
4. KM-ETCS-ID1 Read the ETCS-ID expanded of destination KMC entity of KMAC aborted message, Verify that the destination KMC identity is correct (= the receiver KMC) and store as issuer KMC identity in structure of negative acknowledgment,
5. KM-ETCS-ID2 Read the ETCS-ID expanded of issuer KMC entity of KMAC aborted message and store as destination KMC identity in structure of negative acknowledgement,

6. ISSUE-DATE Use current date and store in structure,
7. TNUM Use the TNUM of the KMAC aborted transaction and store in structure of negative acknowledgment,
8. REASON Define reason for KMAC negative acknowledgment and store in structure,
9. CBC-MAC Calculate CBC-MAC using K-KMC1 and store result in structure of negative acknowledgment,

8.4.5.2 These actions shall be performed for each single KMAC negative acknowledgment between KM interoperable domains.

8.4.5.3 The following table describes the format of KMAC negative acknowledgment :

Octet	Bit 7654 3210	Field name	Field
1	0000 0000	KMAC-NEGACK	KMAC negative acknowledgment message.
2	xxxx xxxx	AB-MESSAGE	Codification of the aborted KM message. Authorized values are KMAC-EXCHANGE, KMAC-DELETION or KMAC-UPDATE
3 4 5 6	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	OB-ETCS-ID	The ETCS-ID expanded of the On-Board ERTMS entity (e.g. an OBU ETCS-ID).
7 8 9 10	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID1	The ETCS-ID expanded of the issuer KMC ERTMS entity of the KMAC update confirmation
11 12 13 14	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	KM-ETCS-ID2	The ETCS-ID expanded of the destination KMC ERTMS entity of the KMAC update confirmation
15 16 17	xxxx xxxx xxxx xxxx xxxx xxxx	ISSUE-DATE	The date of production of the KM negative acknowledgment (in DD MM YY format e.g. 01 01 05 and BCD coded)
18	xxxx xxxx	TNUM	This Transaction Number identifies unambiguously the negative acknowledgment and is equal to the TNUM received in the aborted KMAC message.
19	xxxx xxxx	REASON	Prescribed values are : 0000 0001 Invalid CBC-MAC calculation of received KMAC message 0000 0010 Unknown OBU ETCS-ID 0000 0011 Invalid KMAC parity 0000 0100 Unknown KMAC (no key matching SNUM and KMC Id) Note : other values are reserved for bilateral agreement in KMC agreement
20 21 22 23 24 25 26 27	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	CBC-MAC	The CBC-MAC shall be calculated over the message from octet 1 up to but excluding the CBC-MAC field. The CBC-MAC shall be calculated using K-KMC1 and according to §6.4.2.

Table 14 Structure Information of KMAC negative acknowledgment

END OF DOCUMENT