**ISO 20022**

Card Payments Exchanges (CAPE) - Terminal Management - Maintenance 2024-2025

Message Definition Report Part 1

Approved by the Cards and Related Retail Financial Services SEG on 20 February 2025

This document provides information about the use of the messages Card Payments Exchanges (CAPE) - Terminal Management and includes, for example, business scenarios and messages flows.

March 2025

Table of contents

[1. Introduction 4](#_Toc192178050)

[1.1 Terms and definitions 4](#_Toc192178051)

[1.2 Glossary 4](#_Toc192178052)

[1.3 Document Scope and Objectives 6](#_Toc192178053)

[1.4 References 6](#_Toc192178054)

[2. Scope and Functionality 7](#_Toc192178055)

[2.1 Background 7](#_Toc192178056)

[2.2 Scope 7](#_Toc192178057)

[2.3 Groups of MessageDefinitions and Functionality 7](#_Toc192178058)

[3. BusinessRoles and Participants 8](#_Toc192178059)

[4. BusinessProcess Description 10](#_Toc192178060)

[4.1 BusinessProcess Diagram 10](#_Toc192178061)

[4.2 BusinessProcess Process Flows 11](#_Toc192178062)

[5. Description of BusinessActivities 12](#_Toc192178063)

[5.1 Generic Terminal Management Process 13](#_Toc192178064)

[6. BusinessTransactions 15](#_Toc192178065)

[6.1 Introduction 15](#_Toc192178066)

[6.2 Configuration initiated manually by the acceptor 16](#_Toc192178067)

[6.3 Configuration initiated by a card payment transaction 17](#_Toc192178068)

[6.4 Configuration initiated by a cyclic call 18](#_Toc192178069)

[6.5 Configuration per file download 20](#_Toc192178070)

[6.6 Rejection of a StatusReport 21](#_Toc192178071)

[6.7 Cryptographic key injection in the POI 22](#_Toc192178072)

[6.8 Software Update of a POI 24](#_Toc192178073)

[6.8.1 Through a legacy protocol. 24](#_Toc192178074)

[6.8.2 Through ISO20022 protocol. 25](#_Toc192178075)

[6.9 Creation of a Maintenance Delegation for Key Download 26](#_Toc192178076)

[6.10 Creation of an X.509 Certificate 28](#_Toc192178077)

[7. Revision Record 29](#_Toc192178078)

**Preliminary note:**

The Message Definition Report (MDR) is made of three parts:

* **MDR - Part 1** describes the contextual background required to understand the functionality of the proposed message set. Part 1 is produced by the submitting organisation that developed or maintained the message set in line with a MDR Part1 template provided by the ISO 20022 Registration Authority (RA) on [www.iso20022.org](http://www.iso20022.org/)
* **MDR – Part 2** is the detailed description of each message definition of the message set. Part 2 is produced by the RA using the model developed by the submitting organisation.
* **MDR – Part 3** is an extract of the ISO 20022 Business Model describing the business concepts used in the message set. Part 3 is an Excel document produced by the RA.

# Introduction

## Terms and definitions

The following terms are reserved words defined in ISO 20022 – Part1. When used in this document, they will follow the UpperCamelCase notation.

|  |  |
| --- | --- |
| Term | Definition |
| BusinessRole | functional role played by a business actor in a particular BusinessProcess or BusinessTransaction |
| Participant | involvement of a BusinessRole in a BusinessTransaction |
| BusinessProcess | unrealized definition of the business activities undertaken by BusinessRoles within a BusinessArea whereby each BusinessProcess fulfils one type of business activity and whereby a BusinessProcess may include and extend other BusinessProcesses |
| BusinessTransaction | particular solution that meets the communication requirements and the interaction requirements of a particular BusinessProcess and BusinessArea |
| MessageDefinition | formal description of the structure of a MessageInstance |

## Glossary

**Acronyms**

|  |  |
| --- | --- |
| Abbreviation | Definition |
| AES | Advanced Encryption Standard |
| ‍API | Application Programming Interface |
| ASN.1 | Abstract Syntax Notation 1 |
| CAPE | CArd Payment Exchanges |
| catm | CArd Terminal Management |
| DES | Data Encryption Standard |
| DUKPT | Derived Unique Key Per Transaction |
| EMV | Europay, MasterCard, Visa |
| FIPS | Federal Information Processing Standard |
| FTP | File Transfer Protocol |
| ICC | Integrated Circuit Card |
| IP | Internet Protocol |
| ISO | International Organization for Standardization |
| ‍JSON | Javascript Object Notation |
| KCV | Key Check Value |
| KEK | Key Encryption Key |
| MAC | Message Authentication Code |
| MDR | Message Definition Report |
| MTM | Master Terminal Manager |
| PAN | Primary Account Number |
| PED | PIN Entry Device |
| PKCS | Public Key Cryptographic Standard |
| PKI | Public Key Infrastructure |
| POI | Point Of Interaction |
| POS | Point Of Sales |
| PSP | Payment Service Provider |
| RFID | Radio Frequency Identification |
| RSA | Rivest Shamir Adleman |
| SHA | Secure Hash Algorithm |
| TM | Terminal Manager |
| TMS | Terminal Management System |
| UKPT | Unique Key Per Transaction |
| UML | Unified Modelling Language |
| XML | eXtensible Mark-up Language |

## Document Scope and Objectives

This document is the first part of the ISO 20022 Message Definition Report (MDR) that describes the BusinessTransactions and underlying message set. For the sake of completeness, the document may also describe BusinessActivities that are not in the scope of the project.

This document sets:

* The BusinessProcess scope (business processes addressed or impacted by the project)
* The BusinessRoles involved in these BusinessProcesses

The main objectives of this document are:

* To explain what BusinessProcesses and BusinessActivities these MessageDefinitions have addressed
* To give a high level description of BusinessProcesses and the associated BusinessRoles
* To document the BusinessTransactions and their Participants (sequence diagrams)
* To list the MessageDefinitions

## References

| Document | Version | Date | Author |
| --- | --- | --- | --- |
| ISO 20022 Business Justification – Card Payments Exchanges (CAPE) – RA ID #20 | 2009 | 06 Oct. 2009 | Nexo |
| Terminal Management Message Usage Guide[[1]](#footnote-1) | 14.0  a.k.a 2025 |  | Nexo |
| Card Payment Protocols Security[[2]](#footnote-2) | 4.0 |  | Nexo |
| CAPE Maintenance 2024/2025 – MCR ID #250 | 3.0 | 30 Sept. 2024 | Nexo |

# Scope and Functionality

## Background

This Message Definition Report covers a set of 8 candidate ISO 20022 MessageDefinitions developed by Nexo in close collaboration with the card payment industry. These MessageDefinitions are specifically designed to support card payment terminal management.

## Scope

The messages can be used for the management of card payment terminals or a complete payment system at the merchant.

## Groups of MessageDefinitions and Functionality

These management services are supported by the following messages.

**Monitoring, Maintenance and Configuration messages:**

* The *StatusReport* (catm.001): sent to the master terminal manager or delegated terminal manager to inform them about the status of the POI or values recorded by some probes on the POI;
* The *ManagementPlanReplacement* (catm.002): sent to the POI to provide maintenance actions to be performed;
* The *AcceptorConfigurationUpdate* (catm.003): sent to the POI to transfer configuration parameters from the terminal manager to the POI. These parameters could be for various aims like for instance relative to a payment application or to setup communications between actors by providing network configuration, encoding mode and type of exchange;

**Rejection of received messages:**

* The *TerminalManagementRejection* (catm.004): sent from a POI, a MTM or a TM when the recipient cannot interpret or process the incoming message;

**Maintenance delegation messages:**

* The *MaintenanceDelegationRequest* (catm.005): sent by a terminal manager to the master terminal manager to request the delegation of a maintenance function or maintenance actions;
* The *MaintenanceDelegationResponse* (catm.006): sent by the master terminal manager to provide the outcome of a delegation requested by a terminal manager;

**Certificate management messages:**

* The *CertificateManagementRequest* (catm.007): sent by a POI to a terminal manager acting as a certificate authority for managing X.509 certificate of a public key owned by the POI;
* The CertificateManagementResponse (catm.008): sent by the terminal manager to provide the outcome of a certificate service requested by the POI.

# BusinessRoles and Participants

A BusinessRole represents an entity (or a class of entities) of the real world, physical or legal, a person, a group of persons, a corporation. Examples of BusinessRoles: “Financial Institution”, “ACH”, “CSD”.

A Participant is a functional role performed by a BusinessRole in a particular BusinessProcess or BusinessTransaction: for example the “user” of a system, “debtor”, “creditor”, “investor” etc.

The relationship between BusinessRoles and Participants is many-to-many. One BusinessRole (that is, a person) can be involved as different Participants at different moments in time or at the same time: "user", "debtor”, "creditor", "investor", etc. Different BusinessRoles can be involved as the same Participant.

In the context of payment terminal management, the high-level BusinessRoles and typical Participants can be represented as follows.



| **Participants and BusinessRoles definitions** | |
| --- | --- |
| **Description** | **Definition** |
| Participants | |
| (Account) Servicer (Role) | Party that manages the account on behalf of the account owner (that is, manages the registration and booking of entries on the account, calculates balances on the account and provides information about the account), or the party that has a contractual relationship with the owner (for example, market data provider). |
| (Account) Owner (Role) | Party that legally holds the account or the party in a contractual relationship with the servicer. |
| BusinessRoles | |
| Global Custodian | The party that safekeeps and administers assets on behalf of the owner and that has a network of sub-custodians. |
| Sub-Custodian | The party that safekeeps and administers assets on behalf of the owner |
| Stock Exchange | Place of trade identified with Market Identifier Code. |
| Issuer | Legal Entity that has the right to issue securities |
| Issuer Agent | Organisation appointed by the issuer for the purposes of administration of a security issue or processing of a corporate action or a meeting event. In some cases, the issuer acts as its own agent. |
| CSD | An infrastructure that, holds or controls, the holding of physical or dematerialised financial instruments belonging to all, or a large portion of, the investors in a securities market. This affects the centralised transfer of ownership of such securities by entries on its books and records owner. |
| ICSD | Holds or controls the holding of physical or dematerialised financial instruments belonging to all or a large portion of the investors in a securities market, and which effects the centralised transfer of such securities against payment (or free of payment) by entries on its books and records, resulting in transfer and record of ownership of the securities. A distinction can be made between a Central Securities Depository (CSD), which operates in a particular domestic market for specified financial instruments or an International Central Securities Depository (ICSD), that is, Euroclear and Clearstream, which settle Eurobond instruments and have direct or indirect links with specific CSDs owner. |
| Market Data Provider | A source of financial information. It provides financial news and data (for example, facts, statistics and analysis), for professional and individual investors through various media (for example, the Internet, magazine). |
| Institutional Investor | An organization whose primary purpose is to invest its own assets or those held in trust by it for others. Includes [investment companies](http://www.investorwords.com/2609/investment_company.html), [mutual funds](http://www.investorwords.com/3173/mutual_fund.html), [brokerages](http://www.investorwords.com/585/brokerage.html), [insurance companies](http://www.investorwords.com/6843/insurance_company.html), [pension funds](http://www.investorwords.com/3652/pension_fund.html), [investment banks](http://www.investorwords.com/2602/investment_bank.html). |
| Fund Manager | Entity that implements the investment strategy, that is, selects portfolio investments in accordance with the objectives and strategy in the fund's prospectus, and places orders to effect or liquidate selected investments in accordance with net flow of capital into or out of the fund. |
| Broker | The party that provides services to its members for the settlement of transactions and holding of assets. |
| Beneficial Owner | The person(s) or entity entitled to the benefits of ownership even though another party such as a broker or bank--the nominal owner--actually has possession and title to the security. (Source: [http://www.rbeck.com](http://www.rbeck.com/)). |

|  |  |  |
| --- | --- | --- |
| **BusinessRoles/Participants Matrix Table** | | |
| Participants  BusinessRoles | (Account) Servicer | (Account) Owner |
| Global Custodian | X | X |
| Sub-Custodian | X | X |
| Stock Exchange | X |  |
| Clearing Agent | X |  |
| ICSD | X | X |
| CSD | X | X |
| Market Data Provider | X |  |
| Institutional Investor |  | X |
| Fund Manager |  | X |
| Broker | X | X |
| Beneficial Owner |  | X |

# BusinessProcess Description

## BusinessProcess Diagram

This diagram pictures the high level BusinessProcesses covered by payment terminal management.



**Terminal management process:**

* *Definition*: The process of managing a POI terminal or POI system.
* *Trigger*: The process is triggered manually, by a remote request, or periodically.
* *Pre-conditions*: The POI is initialised.
* *Post-conditions*: The terminal manager and POI are updated.
* *Roles*: POI, (master) terminal manager.

## BusinessProcess Process Flows

The process flows hereafter describe a high level sequence of the BusinessProcess defined in the previous chapter.



# Description of BusinessActivities

This section presents the different BusinessActivities within each BusinessProcess. BusinessActivities of a process are described in swim lane diagrams and are referred in this document as activity diagrams.

The development of an activity diagram is part of the ISO 20022 modelling process and allows capturing the requirements.

The activity diagram provides a zoom-in on the BusinessActivities taking place during each of the BusinessProcesses described in Section 4. It also shows the BusinessActivities that are triggered when another BusinessActivity has a negative result.

What is the activity diagram about?

* It is a diagram representing the ‘common lifecycle’ of a BusinessProcess
* A start point  shows where the lifecycle of the BusinessProcess commences and the end points show End point where the lifecycle may possibly end
* A lozenge means that a choice between several actions can be made
* A bar means that several actions are initiated in parallel
* The flow of activities between the involved Participants (parties)
* BusinessActivities may result in different actions, that is, information is conveyed from one party to another party.

Both in-scope and out-of-scope activities are included, with a different level of details. There are no information requirements for out-of-scope activities, except that they should be clearly identified in the diagram.

Activity diagrams are always accompanied with a text describing the BusinessActivities and their interactions.

## Generic Terminal Management Process



|  |  |
| --- | --- |
| **Descriptions of the BusinessActivities** | |
|  | **Initiator** |
| **Prepare status**: The POI prepares the information to send a StatusReport message, which includes:   * the status of each component, * the events and errors occurred in the system, * the identification of the data set requested in return, if any. | **POI** |
| **Contact Maintenance**: A session with the terminal manager is opened with the information provided by:   * the maintenance action, or * the response to a card payment message (caaa), or * the operator requesting the contact to the maintenance. | **POI** |
| **Identify the maintenance actions**: Depending on the status of the POI and the needs of the terminal manager, no, one or several maintenance operations have to be performed by the POI. | **Terminal Manager** |
| **Replace the management plan**: If the terminal manager has sent a new management plan, the POI replaces the current one for this terminal manager by the new one. | **POI** |
| **Perform local action**: The terminal manager perform locally the initiated maintenance operation of the management plan. | **POI** |
| **Prepare the parameters**: Depending on the status of the POI and the needs of the terminal manager, the terminal manager prepares the requested parameters. | **Terminal Manager** |
| **Store the parameters**: The POI stores the data set sent by the terminal manager in the *AcceptorConfigurationUpdate* message. | **POI** |

# BusinessTransactions

This section describes the message flows based on the activity diagrams documented above. It shows the typical exchanges of information in the context of a BusinessTransaction.

## Introduction

The following collection of terminal management BusinessTransactions shows how a terminal manager is informed by the status of a POI and in which way the terminal manager is able to install and update configuration data of the POI. These BusinessTransactions are typical ones, but this is not an exhaustive list.

These BusinessTransactions are supported by the upload and download of separate terminal management messages.

The terminal management session is always initiated by the POI. The POI starts the session caused by three different mechanisms:

1. The acceptor may start the terminal management session manually using a maintenance function of the POI.
2. The acquirer host sends a trigger in any card payment response message to the POI. The trigger contains the information how and when to contact a TMS. The POI uses this information to establish the connection to the given TMS.
3. The cyclic call is a BusinessTransaction to request the POI to initiate a terminal management session on a regular time basis. This trigger is performed by a specific action in the management plan of the POI.

Additional BusinessTransactions describe the maintenance sessions of the POI with additional terminal managers. These maintenance operations have been delegated to these additional terminal managers by the master terminal manager.

The management plan is used to configure the management actions to be performed by the POI. It contains a list of terminal management actions to be initiated by the POI. The management plan parameters may include for each action: a start time, a delay between the management sessions, and either a limited number of sessions or an end time.

The exchanged terminal management messages are always secured by cryptographic algorithms. The necessary cryptographic keys are pre-installed in the firmware of the POI. Before accepting a message the receiver has to verify the signature of the message generated by the sender.

## Configuration initiated manually by the acceptor



The acceptor uses a maintenance function of the POI to establish a communication session with the terminal manager. The acceptor may choose one of the available terminal managers in the maintenance menu of the POI.

1. The POI sends the *StatusReport* to inform the terminal manager about its parameter version.
2. The terminal manager responds with a *ManagementPlanReplacement* containing the list of actions to be performed including the action for the download of acquirer parameters.
3. The POI sends a *StatusReport* to request the new version of acquirer parameters according to the required action.
4. The terminal manager sends back an *AcceptorConfigurationUpdate* containing the new version of the acquirer parameters.
5. The POI sends a *StatusReport* to inform the terminal manager about the result of the configuration update according to an action of the management plan.
6. The terminal manager responds with a *ManagementPlanReplacement* without any action.

## Configuration initiated by a card payment transaction



1. The acceptor sends to the acquirer its version of the acquirer parameters[[3]](#footnote-3) in the *AcceptorAuthorisationRequest* message.
2. The acquirer detects that the parameters have expired. The acquirer sends the *AcceptorAuthorisationResponse* message, requesting to contact the terminal manager.
3. The POI sends the *StatusReport* to inform the terminal manager about its parameter version.
4. The terminal manager responds with a *ManagementPlanReplacement* containing the list of actions to be performed including the action for the download of acquirer parameters.

The BusinessTransaction continues with the updating of the acquirer parameters as described in section 6.2 (as from step 3).

## Configuration initiated by a cyclic call

In this BusinessTransaction the management plan of the terminal manager contains an action to perform a cyclic call.

A cyclic call is an action to request the POI system to contact the terminal manager on a regular time basis.



1. The POI opens a maintenance session when the trigger conditions of the cyclic call are met. The POI sends the *StatusReport* to inform the terminal manager about the status of the POI.
2. The terminal manager responds with the *ManagementPlanReplacement* containing no actions to be performed.
3. The POI waits for opening a new session until the trigger conditions of this cyclic call are met. The POI sends then again the *StatusReport* to inform the terminal manager about the status of the POI.
4. An update of the acquirer parameters is now required, then the terminal manager generates a management plan containing actions for:

* download parameters,
* a cyclic call.

1. When the start time of the action is reached, the POI contacts the terminal manager sending a *StatusReport* requesting a configuration update.
2. The terminal manager responds with the *AcceptorConfigurationUpdate* message containing the parameters.
3. At the next cyclic call the POI issues a *StatusReport* containing the final version of the installed parameter sets as well as logging information about the successful or failed management functions.
4. The terminal manager responds with a *ManagementPlanReplacement* message with no action. The POI keeps the current management plan to continue the cyclic call.

## Configuration per file download



This BusinessTransaction is identical to the BusinessTransaction presented in setion 6.2, replacing the message exchange transport protocol by file transfer protocol.

1. The POI uploads the *StatusReport* file to inform the terminal manager about its parameter version.
2. The POI downloads a new *ManagementPlanReplacement* file generated by the terminal manager containing two actions:

* Download parameters, and
* Upload status.

1. The POI downloads the *AcceptorConfigurationUpdate* file and installs the parameters.
2. The POI uploads a *StatusReport* file with the versions of the installed acquirer parameter.

## Rejection of a StatusReport

A rejection is sent by a terminal manager to a POI to indicate that the received message could not be processed by the terminal manager (e.g. unable to read or process the message, security error, duplicate message, etc.).



1. The terminal manager cannot process the *StatusReport* sent by the POI.
2. The acquirer sends a *TerminalManagementRejection* in place of any response message (*ManagementPlanReplacement* or *AcceptorConfigurationUpdate*).

## Cryptographic key injection in the POI

In this BusinessTransaction the terminal manager performs cryptographic key injection in a POI terminal with the following assumptions:

* The terminal manager owns an asymmetric key pair KeyTM-SIG to perform digital signatures of messages, certified by the certificate CertTM-SIG.
* The POI owns an asymmetric key pair KeyPOI-ENC for encryption, to protect the transport of the session key, certified by the certificate CertPOI-ENC.
* The POI owns an asymmetric key pair KeyPOI-SIG for authentication, to perform digital signatures of messages, certified by the certificate CertPOI-SIG.



1. The POI opens a maintenance session with the terminal manager, sending a *StatusReport* containing the status of the cryptographic keys stored in the POI, the POI certificates for signing and enciphering and asking for the TM certificate.
2. The terminal manager responds with an *AcceptorConfigurationUpdate*. The message is digitally signed with the asymmetric key KeyTM-SIG.  
   This action includes a fresh challenge (TM challenge) generated by the terminal manager and its public key for signature KeyTM-SIG within the certificate CertTM-SIG.
3. The POI sends a *StatusReport* requesting the update of parameters. The message is digitally signed with the asymmetric key KeyPOI-SIG.  
   The message contains the previous terminal manager challenge (TM challenge), a challenge (POI challenge) generated by the POI.
4. The terminal manager sends a *AcceptorConfigurationUpdate* to inject the new cryptographic keys. The message is digitally signed with the asymmetric key KeyTM-SIG. This message includes:

* the challenge generated by the terminal manager (TM challenge),
* the challenge sent by the POI (POI challenge), and
* the cryptographic keys to inject in the POI. Keys are enciphered by with a KEK key. The KEK key is enciphered by KE and KE is enciphered with the asymmetric KeyPOI-ENC

1. The POI decrypts and loads the new keys, computes a key check value (KCV) of keys, and sends a *StatusReport* with the new status of the loaded keys and the KCV.   
   This message is digitally signed with the asymmetric key KeyPOI-SIG.
2. This status report message confirms to the terminal manager the injection of the cryptographic keys in the POI. The terminal manager responds with a *ManagementPlanReplacement* containing no actions.

## Software Update of a POI

### Through a legacy protocol.



1. The POI opens a maintenance session when the trigger conditions of the cyclic call are met. The POI sends the *StatusReport* to inform the master terminal manager about the status of the POI.
2. The master terminal manager responds with the *ManagementPlanReplacement* containing the maintenance actions including an action to perform update of software modules from a terminal manager host, using a dedicated protocol.
3. The POI performs the action, contacting the terminal manager host with the provided information, to update the software modules.
4. The POI issues a *StatusReport* to the master terminal manager to inform about the result of the software update.
5. The terminal manager responds with a *ManagementPlanReplacement* message with no action.

### Through ISO20022 protocol.



1. The POI opens a maintenance session when the trigger conditions of the cyclic call are met. The POI sends the *StatusReport* to inform the terminal manager about the status of the POI.
2. The terminal manager responds with the *ManagementPlanReplacement* containing the maintenance actions including an action to perform update of software modules optionally from a different terminal manager host.
3. The POI sends a *StatusReport* to request the new version of software modules according to the required action.
4. The master terminal answers with the new software modules included in an *AcceptorConfigurationUpdate.*
5. The POI issues a *StatusReport* to the terminal manager to inform about the result of the software update.
6. The terminal manager responds with a *ManagementPlanReplacement* message with no action.

## Creation of a Maintenance Delegation for Key Download



The terminal manager and the master terminal manager negotiate the delegation of key download to define the scope of the delegation, the subset of the terminal estate, and the identifiers to be used.

1. The terminal manager sends a *ManagementDelegationRequest* to the master terminal manager of the terminal estate with the scope of the key download and potentially identification of the keys.
2. The master terminal manager create a new delegation of key download for this new terminal manager, and sends a *ManagementDelegationResponse* to the terminal manager to inform that the delegation is taken into account for the terminal estate.
3. A POI opens a maintenance session sending a *StatusReport* to inform the master terminal manager about the status of the POI.
4. The POI belongs to the subset of terminals for which the delegation has been created. The master terminal manager responds with the *ManagementPlanReplacement* containing the maintenance action for the creation of the delegation. This action contains the scope of the delegation, the information to identify and contact the delegated terminal manager, and the proof to provide to the terminal manager.
5. The POI contacts the delegated terminal manager by sending a *StatusReport* to inform this terminal manager it belongs to the subset of the terminal estate which is managed by the delegated terminal manager for the key download.
6. The delegated terminal manager, validates the proof provided by the POI, and responds with a *ManagementPlanReplacement* dedicated to the delegated terminal manager. The management plan contains action to download cryptographic keys, and key download will start with the delegated terminal manager.
7. At the next contact with the master terminal manager, the POI issues a *StatusReport* containing information about the successful of the delegation creation with the new terminal manager.
8. The terminal manager responds with a *ManagementPlanReplacement* message with no action.

## Creation of an X.509 Certificate



A POI terminal requests the creation of an X.509 certificate with the public key and the information of the owner of the asymmetric key provided by the POI.

1. The POI sends a *CertificateManagementRequest* to the terminal manager, containing the public key of the asymmetric key and the information about the key owner, using the PKCS 10 format. The message is digitally signed with any asymmetric key recognised by the terminal manager.
2. After validation of the request message, the terminal manager generates an X.509 certificate with the public key and the attributes provided in the request, and sends it to the POI terminal in a *CertificationManagementResponse*.   
   The POI terminal stores the certificate with the path to the root of the terminal manager.

# Revision Record

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Revision** | **Date** | **Author** | **Description** | **Sections affected** |
| 1.0 | December 2024 | Nexo | Draft version for SEG review | All |
| 2.0 | March 2025 | ISO 20022 RA | Approved version | All |
|  |  |  |  |  |

**Disclaimer**:

Although the Registration Authority has used all reasonable efforts to ensure accuracy of the contents of the iso20022.org website and the information published thereon, the Registration Authority assumes no liability whatsoever for any inadvertent errors or omissions that may appear thereon. Moreover, the information is provided on an "as is" basis. The Registration Authority disclaims all warranties and conditions, either express or implied, including but not limited to implied warranties of merchantability, title, non-infringement and fitness for a particular purpose.

The Registration Authority shall not be liable for any direct, indirect, special or consequential damages arising out of the use of the information published on the iso20022.org website, even if the Registration Authority has been advised of the possibility of such damages.

**Intellectual Property Rights**:

The ISO 20022 MessageDefinitions described in this document were contributed by nexo-standards. The ISO 20022 IPR policy is available at www.ISO20022.org > About ISO 20022 > Intellectual Property Rights.

1. Available on www.nexo-standards.org [↑](#footnote-ref-1)
2. Available on www.nexo-standards.org [↑](#footnote-ref-2)
3. See the MDR of the caaa business area for the description of the *AcceptorAuthorisationRequest* and *AcceptorAuthorisationResponse* messages. [↑](#footnote-ref-3)