Digital Tachograph
Specification for remote company card authentication and remote data downloading.

Version 02.01

Classification	Distribution		Distribution	
	Name	Comp./Dept.	Name	Comp./Dept.
☒ No ☐ Company Confidential ☐ Customer Confidential ☐	Members of Heavy Truck Electronic Interfaces Working Group			

	By	Date	Sign			Index
Prepared by	ACEA-HDEI			Digital Tachograph : specification for remote company card authentication remote data downloading	Vers. 02.01	09.12.2018
	TF remote download	18/12/09				
					Page 1 / 69	Format A4

Issue	Pages/section	Date	Evolutions
-	All	15/09/09	Document creation – publication on www.fms-standard.com
-	IV.1	14/09/12	Text reorganised to introduce CAN wake-up capability
-	All	29/10/18	Text updated in accordance with new Tachograph Regulation
-	IV.3, V.2.2.1.3, V.2.2.4	29/10/18	Updates related to new TREPs 0x21 to 0x25 for tachograph generation 2 VU data format
-	All	09/12/18	Publication on www.fms-standard.com

TABLE OF CONTENTS

I. PRESENTATION

I.1 DOCUMENT SUBJECT

This document specifies the procedures to be followed by a digital tachograph, in order to perform:
- initial authentication of a remote company card,
- remote downloading of tachograph and driver card data.

I.2 APPLICATION DOMAIN

This document is based on the results of the standardisation works of the Heavy Truck Electronic Interfaces Working Group, and its purpose is to provide third parties with information they need to design other parts of the complete system allowing a company to download VU data from distant vehicles, such as:
- on-board units,
- back office software and associated tools.

I.3 DOCUMENT DESCRIPTION

Chapters I, II and III present this document, provide the list of related documents, and the useful terminology.

Chapter IV is a functional specification of the remote card authentication and data download procedures.

Chapters V, VI, VII, VIII detail the application protocol, the network layer, the data link layer, and the physical layer, respectively.

Annex 1 provides examples of message sequences between the FMS and the VU.
Annex 2 provides user guidance for managing error messages received from the VU.
Annex 3 provides user guidance for managing the company card reader (T0 and T1 protocols)

I.4 TYPOGRAPHIC CONVENTIONS

None.

II. DOCUMENTS

II.1 APPLICABLE DOCUMENTS

None.

II.2 REFERENCE DOCUMENTS

[Annex 1C]	Annex1C of Regulation (EU) n°2016/799 and particularly:
[Annex1C Main Body]	Main Body
[Annex1C Appendix 1]	Data Dictionary
[Annex1C Appendix 7]	Data Downloading Protocol
[Annex1C Appendix 8]	Calibration Protocol
[Annex1C Appendix 11]	Common Security Mechanisms
[Annex1C Appendix 15]	Migration
[VU PP]	BSI-CC-PP-0094, Common Criteria Protection Profile – Digital Tachograph – Vehicle Unit (VU PP) V1.0
[Annex1B]	Annex1B of Modified Regulation 3821/85/EEC, and particularly:
[Annex1B Main Body]	Main Body
[Annex1B Appendix 1]	Data Dictionary
[Annex1B Appendix 7]	Data Downloading Protocol
[Annex1B Appendix 8]	Calibration Protocol
[Annex1B Appendix 10]	Generic Security Targets
[Annex1B Appendix 11]	Common Security Mechanisms

[ISO 14229-1]	ISO14229-1:2005 (dated 2006-01-10 Road vehicles – Unified diagnostic services – Part 1 : Specification and requirements)
[ISO 15765-2]	ISO/FDIS15765-2 Road vehicles – Diagnostics on CAN – Part 2 : Network layer services
[ISO 15765-3]	ISO/FDIS15765-3 Road vehicles – Diagnostics on CAN – Part 3 : Implementation of unified diagnostic services
[ISO 16844-1]	ISO/FDIS16844-1 Road vehicles – Tachograph systems – Part 1 : Electrical connectors
[ISO 16844-4]	ISO/FDIS16844-4 Road vehicles – Tachograph systems – Part 4 : CAN Interface
[ISO 16844-6]	ISO/FDIS16844-6 Road vehicles – Tachograph systems – Part 6 : Diagnostics
[ISO 16844-7]	ISO/FDIS16844-7 Road vehicles – Tachograph systems – Part 7 : Parameters
[ISO7816-3]	ISO7816-3 Identification cards – Integrated circuit(s) cards with contacts – Part 3 : Electronic signals and transmission protocols
[ISO 7816-4]	ISO7816-4 Identification cards – Integrated circuit(s) cards with contacts – Part 4 : Organization, security and commands for interchange

III. TERMINOLOGY

III.1 ABBREVIATIONS

APDU	Application Protocol Data Unit
ATR	Answer To Reset (as defined in [ISO7816-3])
ESM	External Storage Medium (as defined in [Annex1B Appendix 7])
FMS	Fleet Management System
IDE	Intelligent Dedicated Equipment (as defined in [Annex1B Appendix 7])
LSB	Least Significant Byte
MSB	Most Significant Byte
N_PDU	Network Protocol Data Unit
SID	Service identifier (as defined in [Annex1B Appendix 7] / [ISO14229-1])
STmin	Separation time min. (between the transmission of 2 ConsecutiveFrames protocol data units, as defined in [ISO15765-2])
TA3	a character of the Answer To Reset of a tachograph card (as defined in [ISO7816-3])
Tauth	Authentication Timeout
Trem	Timeout for each transaction between the VU and the remote company and vice versa
TREP	Transfer response parameter (as defined in [Annex1B Appendix 7] / [ISO14229-1])
TRTP	Transfer request parameter (as defined in [Annex1B Appendix 7] / [ISO14229-1])
VU	Vehicle Unit (as defined in [Annex1B]), also called digital tachograph in this document

III.2 GLOSSARY

Mutual authentication protocol	The mutual authentication protocol between a remote company card and a VU, according to [Annex1C Appendix 11], requirement CSM_020, sections 10.1 to 10.5, or [Annex1B Appendix 11], requirement CSM_020.
(Remote) authentication process	The remote authentication process includes the mutual authentication protocol, followed by the transfer of the list of data required by the Fleet Back Office to the VU. After a successful authentication process, access to the required data is granted by the VU.
Valid (remote) authentication	A remote authentication is valid after a remote authentication process has been successful, and while the Tauth Authentication Timeout is active.
(Remote) download process, (remote) data transfer	The remote download process (or remote data transfer) includes all steps necessary to transfer data required by a remotely authenticated company, while its authentication remains valid.
Gen1 VU, Gen1 Tachograph Cards	compliant with Annex1B of Modified Regulation 3821/85/EEC.
Gen2 VU, Gen2 Tachograph Cards	compliant with Annex1C of Regulation (EU) n°2016/799.

IV. REMOTE COMPANY AUTHENTICATION AND DATA DOWNLOADING PROCESS

IV.1 OVERVIEW

As specified in the Tachograph Regulation ([Annex1C Appendix 7], [Annex1B Appendix 7]), data may be downloaded from a Vehicle Unit (VU), using an Intelligent Dedicated Equipment (IDE) physically connected to the VU downloading connector. Downloaded data are appended with signatures, in order to give the possibility to verify its authenticity and integrity.

Two generations of Digital Tachographs must be considered:
- Gen1 VU, which are compliant with Annex1B of Modified Regulation 3821/85/EEC,
- Gen2 VU, which are compliant with Annex1C of Regulation (EU) n°2016/799.

Both VU generations will co-exist in the field, so this document addresses Remote tachograph data download in both cases.

The following principles have been set up, to facilitate the migration between both Digital Tachograph Systems (see [Annex1C Appendix 15] for additional details):
- Gen1 Driver and Company Cards can be used in Gen2 VU,
- Gen2 Driver and Company Cards can be used in Gen1 VU.

Gen2 Driver and Company Cards inserted in Gen1 VU behave like Gen1 Cards.

Data is remotely downloaded from Gen1 VU according to the protocol specified in [Annex1B Appendix 7], and from Gen2 VU according to the protocol specified in [Annex1C Appendix 7].

This document addresses therefore two types of remote tachograph data downloads:
- Generation 1 type of VU data download, providing the generation 1 data structure, signed using [Annex1B Appendix 11] Security Mechanisms.
- Generation 2 type of VU data download, providing the generation 2 data structure, signed using [Annex1C Appendix 11 Part B] Security Mechanisms.

Similarly, there are two types of data downloads from driver cards inserted in a Gen1 or Gen2 VU:
- Generation 1 type of card data download, providing the generation 1 data structure, signed using [Annex1B Appendix 11] Security Mechanisms.
- Generation 2 type of card data download, providing the generation 2 data structure, signed using [Annex1C Appendix 11 Part B] Security Mechanisms.

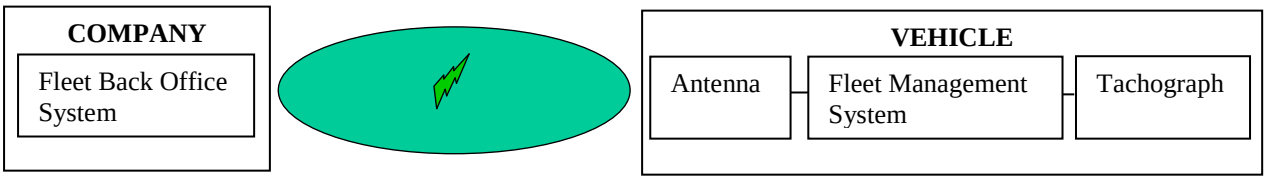
Gen1 cards provide only Generation 1 type data download, while Gen2 cards provide both.

Please note that Gen2 Driver Card data downloaded with a Gen1 VU will only deliver Gen1 Driver Card data.
This might be not accepted by the authorities (is member state dependant).

To download data from the VU, an operator acting on behalf of a company must perform the following steps:
- Insert a company card inside a card slot of the VU,
- Connect the IDE to the VU download connector,
- Establish the connection between the IDE and the VU,
- Select on the IDE the data to download and send the request to the VU;
- Close the download session.

Downloaded data must then be transferred from the IDE to an External Storage Medium (ESM), in order to be available for the company and also for the relevant Control Authorities, as required by applicable national regulation.

The Tachograph Regulation ([Annex1C Main Body], [Annex1B Main Body]) also allows the VU to download data through another connector to a company remotely authenticated through this channel. In such a case, company mode data access rights corresponding to the remotely authenticated company card are applied to the downloaded data.



As illustrated by the above drawing, employees of the transport companies may download tachograph data from remote vehicles, without needing to access directly to the VU.

The Fleet Back Office System is assumed to include a card reader able to communicate with company cards, itself connected to a computer which is able to communicate with the distant vehicles by any method.

As specified by the Tachograph Regulation ([Annex1C], [Annex1B]), the VU may use any one from the two T=0 and T=1 protocols defined by [ISO7816-3], so the card reader shall also be able to support both protocols.

Please note that it is necessary to respond with the protocol the VU is using (not the card)!

Equipment able to manage the communication with the company, and also with the tachograph is also supposed to be installed in the vehicles. The unit communicating with the VU is called Fleet Management System (FMS) in this document.

For remote authentication, a valid company card shall be first inserted in the Fleet Back Office System card reader. Remote authentication and data download shall only be possible if only driver cards or no other valid tachograph cards are inserted in the VU.

Any tachograph card insertion in the VU may lead the VU to terminate a remote card authentication or remote data download process currently in progress. But any remote card authentication or remote data download process shall always be terminated by the VU at the insertion of a valid company, control or workshop card into the VU.

Typically, the company shall start the whole process on its own initiative by sending a request to the vehicle, informing the VU that a remote company card is ready to start a remote authentication process.

Alternately, the FMS in the vehicle can signal to the company that a remote data downloading should be done. When the company is ready, it shall then send a request to the vehicle, informing the VU that a remote company card is ready to start a remote authentication process.

This request shall be received by the VU via the FMS, which can manage simultaneously the communication with the company and the VU.

Seen from the VU point of view, the following steps shall be performed:
- A specific diagnostic session (remoteSession) is opened.
- Authentication process: after the reception of the request indicating to the VU that a remote company card is ready to start an authentication process, the VU starts the remote authentication process, based on the mutual authentication protocol defined in [Annex1C Appendix 11] or [Annex1B Appendix 11]. During the mutual authentication protocol, the VU is the master. It exchanges the appropriate messages with the remote company card. It receives information about the data the company requires to download at the end of the authentication process.
- Data transfer: the data required by the company are downloaded from the VU to the FMS, the FMS having the task to transmit the data to the company. During the VU to FMS download process, the FMS is the master. The downloading protocol is based on the ‘local’ downloading protocol defined in [Annex1C Appendix 7] or [Annex1B Appendix 7].
- The remoteSession is closed.

The FMS shall act as the client and the VU as the server (according to [ISO14229-1] definitions). How data is exchanged between the VU and the FMS is specified further in this document. The communication scheme between the FMS and the VU is based on the following principles:
- the FMS shall open the communication with the VU (if applicable),
- the FMS shall send requests to the VU, possibly including data to be transmitted to the VU,
- the VU shall respond to any request from the FMS with:
- a positive response, possibly including data to transmit to the company,
- or a negative response, including the appropriate error cause,
- The FMS shall close the communication with the VU (if applicable).

This document specifies the messages that shall be used between a VU and a FMS for remote company card authentication and data download.

As specified in section VIII, the remote company card authentication and data download function may be accessed through a dedicated connector, which provides access to a CAN interface of the VU used for both the remote authentication and the remote data download downloading process.

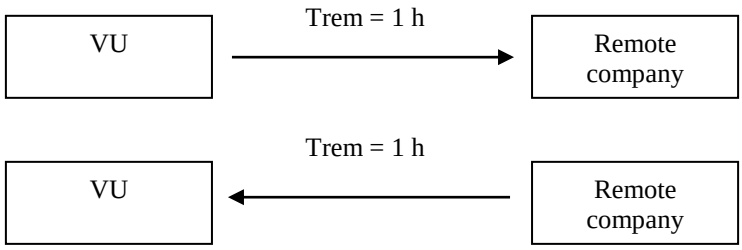
No secured communication on [ISO 14229-1] protocol level is required between the VU and the FMS.

Secured communication is recommended between the vehicle and the remote company (but not mandatory): the purpose is to avoid the divulgation of data outside the company it belongs to.

No interaction is needed between user and VU during remote authentication and download.

During the remote authentication process, the VU manages a time out for each interaction between the VU and the remote company and vice versa (2*Trem). Trem is fixed to 1 hour.

So (2*Trem=2 h) is the maximum allowed time between a response from the VU to the FMS and the next request from the FMS to the VU corresponding to the next step of the authentication process.



If this timeout expires during the remote authentication process, it shall therefore not be handled as unsuccessful authentication by the VU in the meaning of [VU PP], clause 6.1.1.4.2, or [Annex 1B Appendix 10], (marginal UIA_220). But it shall however terminate this authentication process, because of bad communication conditions. A new remote authentication process must then be started from the beginning (RemoteCompanyCardReady message, see IV.2.2).

During ignition off, it shall be possible to perform one or more remote authentications and data downloads.

The capability of remote authentication and download during ignition off is implemented in the VU, in accordance with the following rules:
- the VU is able to finalise an on-going remote authentication and data download at ignition off,
- the VU is able to start new remote authentication(s) and data download(s), but only within 24 hours after ignition off ($Timer1_{CANwake\text{-}up}$),
- the maximum period for remote authentication(s) and data download(s) during ignition is off is 2 hours continuously ($Timer2_{CANwake\text{-}up}$),
- the $Timer2_{CANwake\text{-}up}$ can be started once only within $Timer1_{CANwake\text{-}up}$,
- an on-going remote authentication or data download must not be interrupted by ignition on/off,
- after $Timer1_{CANwake\text{-}up}$ and/or $Timer2_{CANwake\text{-}up}$ has elapsed, an on-going remote authentication or data download can only continue after ignition has been on again.

$Timer1_{CANwake-up}$ and $Timer2_{CANwake-up}$ are specified as:

- $Timer1_{CANwake-up}$:
  - $Timer1_{CANwake-up}$ starts with ignition off
  - $Timer1_{CANwake-up}$ expires due to following conditions:
    - $Timer2_{CANwake-up}$ expires
    - $Timer1_{CANwake-up}$ value reached 24 hours
  - $Timer1_{CANwake-up}$ will be reset with ignition on and is not active during ignition is on
-
- $Timer2_{CANwake-up}$
  - $Timer2_{CANwake-up}$ starts with the first occurrence of one of the following conditions:
    - First start of a remote session during ignition is off (within $Timer1_{CANwake-up}$),
    - On-going remote session at ignition off.
  - $Timer2_{CANwake-up}$ expires due to following conditions:
    - $Timer1_{CANwake-up}$ expires,
    - $Timer2_{CANwake-up}$ value reached 2 hours.
  - $Timer2_{CANwake-up}$ will be reset with ignition on and is not active during ignition is on.

Any communication detected on the CAN shall wake-up the VU CAN transceiver.

Please note that the capability for wake-up for remote authentication is dependent on the tachograph version. Details can be obtained at the manufacturer of the tachograph.

IV.2 AUTHENTICATION OF THE REMOTE COMPANY CARD

IV.2.1 Remote company card authentication process

The authentication process of the remote company card is based on the same mutual authentication protocol as if the card would be inserted in the VU. This protocol is specified in [Annex1B Appendix 11]. Notations and abbreviated terms used in this document are the same as in [Annex1B Appendix 11].

To perform the authentication, a specific diagnostic session (called remoteSession) must be opened by the FMS in the VU.

After a successful remote authentication process, the VU shall allow to download data required by the company while applying data access rights corresponding to the remotely authenticated company (like for the local download specified in [Annex1B]).

After having received from the VU the RemoteDownloadAccessGranted message (see IV.2.2) indicating:
- that the VU's access to the remote company card is no longer needed,
- and that download access has been granted to the data required by the company,

the company may stop the communication with the vehicle.

The VU shall reject any valid authentication and terminate any on-going download through the remote download interface if at least one of the following conditions are met:
- A CloseRemoteAuthentication request is received (see IV.2.2),
- A RequestTranferExit request is received (see IV.3)
- A valid card other than a driver card is inserted in the VU
  (Please note that some VUs are terminating authentication process when inserting any card in one of the VU slots),
- The permanent power supply of the VU is interrupted.

A remote company card shall be authenticated again before a different data download from the already ongoing one through the remote download interface is possible.
However, when one of the above conditions is met, the VU does not automatically quit the remoteSession (except for permanent power supply interruption).

If the withdrawal of an inserted driver card is requested by the user during the data transfer of its card, the card shall be immediately withdrawn. The remote data transfer of the driver card shall be aborted.

In the case the ignition power supply of the VU is cut during the authentication process, and if the communication must be interrupted when ignition is off:
- if any service request of the authentication process was in progress in the VU but not answered at the time of the ignition power loss, the FMS shall send again to the VU the service request corresponding to the step which was not completed (or re-start the complete authentication process),
- if no service request of the authentication process was in progress in the VU at the time of the ignition power loss, the FMS shall send to the VU the service request corresponding to the next step of the authentication process (or re-start the complete authentication process).

An application layer timeout (Tauth=24h) shall therefore be used in the VU for indicating that a remote authentication process is on-going, or a remote company card authentication is valid. Tauth shall limit the cumulated duration of the remote authentication process and data download: any on-going remote authentication process or data download shall only be continued while Tauth is active. Once Tauth has expired or has been closed, a new remote authentication process must be started before any data download. Tauth will be started by the VU when the first message of the remote authentication process is correctly received.

Starting a new card authentication process shall not be accepted by the VU if Tauth is active (which means that another remote authentication process is on-going or another remote authentication is already valid in the VU).

IV.2.2 Messages needed for remote company card authentication

The following table provides details about the requests / responses needed to perform the remote company card authentication process.

All messages used are RoutineControl requests or positive/negative responses, as defined by [ISO14229-1] and this document, with subfunction set to 0x01 (StartRoutine) and routineIdentifier set to 0x0180 (RemoteTachographCardDataTransfer).

Column 1 indicates the direction of the message (->VU means message sent to the VU, VU-> means message sent by the VU)
Column 2 provides the type of the RoutineControl StartRoutine RemoteTachographCardDataTransfer message used (request of positive response)
Column 3 provides the message name
Column 4 provides the RoutineControlOption#1 or RoutineStatus#1 value used
Column 5 details how the message shall be processed
Column 6 details the data included in the message (if any)

Dir.	Message Type	Message Name	ControlOption #1/ Status #1 value	Message process	Data
->VU	Request	RemoteCompanyCardReady	01	This message indicates to the VU that a remote company card is inserted in an appropriate card reader and ready to start the mutual authentication protocol. The ATR of the remote company card is attached.	ATR
VU->	Positive Response	VUReady	02	The VU confirms the readiness for starting the mutual authentication protocol.	
->VU	Request	CompanyCardToVUData	03	The FMS provides a tachograph card APDU from the remote company card to the VU.	None or Tachograph card APDU: Remote company card -> VU
VU->	Positive Response	VUToCompanyCardData	04	The VU sends a tachograph card APDU to the remote company card. CompanyCardToVUData and VUToCompanyCardData provide a data transport for the mutual authentication protocol described in Annex 1B, Appendix 11, CSM_020 “Mutual authentication mechanism”	Tachograph card APDU: VU -> Remote company card
			…	Continue with CompanyCardToVUData and VUToCompanyCardData until RemoteAuthenticationSucceeded.	
VU->	Positive Response	RemoteAuthenticationSucceeded	06	The remote mutual authentication process has succeeded.	
->VU	Request	RemoteDownloadDataRequest	07	The FMS provides the Fleet Back Office download request list to the VU.	Download request list.
VU->	Positive Response	RemoteDownloadAccessGranted	08	The VU grants the download access to the requested data.	
->VU	Request	CloseRemoteAuthentication	09	The Back-Office ends / terminates the remote authentication process. Note: this command can also be used by the company / FMS to enforce to close any valid authentication.	

The following additional positive responses from the VU to the FMS are used to indicate to the company that an error occurred in one of the steps of the authentication process and to inform the company about the possible reason of the error:

VU->	Positive Response	RemoteAuthenticationClosed	0A	The VU confirms that the authentication process is ended / terminated or a previously valid authentication closed.	
VU->	Positive Response	APDUError	0C	The VU informs the company that 3 consecutive APDU errors have occurred.	
VU->	Positive Response	AuthenticationError	0E	The VU informs the company that the card authentication has failed	
VU->	Positive Response	TooManyAuthenticationErrors	10	The VU informs the company that 5 consecutive card authentication errors have occurred	

Please note that the AuthenticationError and the TooManyAuthenticationErrors messages will be sent by the VU if expired remote company cards are used.

IV.3 DATA DOWNLOADING TO THE FMS

Seen from the VU point of view, once the remote authentication process has been successfully completed and access granted to the requested data, the VU is allowed to download data through its link to the FMS. Company mode data access rights corresponding to the remotely authenticated company shall be applied.

Seen from the company point of view, after having received the RemoteDownloadAccessGranted from the VU, the remote authentication process is successfully completed.

The data transferred from the VU to the FMS may be transmitted later to the company by the FMS.

To perform data transfer from the VU to the FMS, a specific diagnostic session (called remoteSession) must be opened by the FMS in the VU. The requests / responses are similar to the ones defined in [Annex1B Appendix 7] for local data downloading, but slightly adapted:
- to be compliant with [ISO14229-1],
- to manage card download from a specified slot (driver or co-driver).

Each request from the FMS shall be followed by one positive or negative response from the VU. All messages are defined in section V.2.

In the case the ignition power supply of the VU is cut during the downloading process, and if the communication must be interrupted when ignition is off:
- if any service request of the downloading process was in progress in the VU but not completely answered at the time of the ignition power loss, the FMS shall send again to the VU the service request corresponding to the service which was not completed (or re-start the complete downloading process),
- if no service request of the downloading process was in progress in the VU at the time of the ignition power loss, the FMS shall send to the VU the service request corresponding to the next service of the downloading process (or re-start the complete downloading process).

After the ignition power supply has been off during a data transfer (if the VU is not able to continue an interrupted data transfer at the point of interruption):
- the FMS sends a StartDiagnosticSession(remoteSession) request,
- the FMS sends a RequestUpload request,
- the FMS sends a TransferData request, specifying the requested data file.

Please note that it might be not possible to restart downloading at the point of interruption. In those cases, the data has to be downloaded from the beginning. However, it is not necessary to restart a remote authentication process (beside permanent power off)
The following table, which is derived from the table in paragraph 2.2.2. of [Annex1B Appendix 7], provides an overview of the messages used. The function associated to each message is the same as the one defined in [Annex1B Appendix 7] for local data downloading.

The required data transfer shall be initialised by the FMS with a RequestUpload service. Then, the required data shall be transferred from the VU to the FMS in several TransferData Positive Responses, each Data field containing 255 bytes maximum.

When the data transfer is completed, the FMS shall send a RequestTranferExit request to the VU, to terminate the remote authentication, so that a new remote card authentication may start.

The data transfer may also be interrupted by the FMS at any time, and resumed later (see V.2.2.1.2).

RequestUpload requests are authorised as many times as needed, while access rights to a DownloadRequestList are open.

Column 1 indicates the direction of the message (FMS ->VU or VU->FMS)
Column 2 provides the name of the message.
Column 3 provides the SID value in the data field of the message.
Column 4 indicates if the blockSequenceCounter is used in the data field of the message.
Column 5 indicates if the wrapAroundCounter is used in the data field of the message (see V.2.2).
Column 6 provides TRTP#2 or TREP#2 value in the data field of the message (if any). TREP#2 21 to 25 are used for response with generation 2 data format.
Column 7 indicates the content of the rest of the data field of the message.

Direction	**Message Name**	**DATA Field**				
		SID	**blockSequence Counter**	**wrapAround Counter**	**TRTP#2/ TREP#2**	**Tranferred Data**
FMS->VU	RequestUpload Request	35				00 44 00 00 00 00 FF FF FF FF
VU->FMS	Positive Response RequestUpload	75				10 FF
FMS->VU	TransferData Request Overview	36	XX	XX	01	
FMS->VU	TransferData Request Activities (Date)	36	XX	XX	02	XX XX XX XX
FMS->VU	TransferData Request Events and Faults	36	XX	XX	03	
FMS->VU	TransferData Request Detailed Speed	36	XX	XX	04	
FMS->VU	TransferData Request Technical Data	36	XX	XX	05	
FMS->VU	TransferData Request Card Download (Slot)	36	XX	XX	06	XX
VU -> FMS	PositiveResponse TransferData Overview	76	XX	XX	01/21	Data
VU -> FMS	PositiveResponse TransferData Activities (1 day)	76	XX	XX	02/22	Data
VU -> FMS	PositiveResponse TransferData Events Faults	76	XX	XX	03/23	Data
VU -> FMS	PositiveResponse TransferData Detailed Speed	76	XX	XX	04/24	Data
VU -> FMS	PositiveResponse TransferData Technical Data	76	XX	XX	05/25	Data
VU -> FMS	PositiveResponse TransferData Card Download (Slot)	76	XX	XX	06	Data
FMS->VU	RequestTransferExit Request	37			00	
VU -> FMS	PositiveResponse RequestTransferExit	77			00	

Notes concerning [Annex1B Appendix 7] requirements for data storage in companies:

- - Any VU data types defined above (Overview, Activities, Events and Faults, Detailed Speed, Technical data, Card Download) transferred from the VU within one download session must be stored within one physical file by the company. Data stored include SID and TREP of the first positive response message if several positive responses are needed, followed by all transferred data, but exclude blockSequence Counter and wrapAroundCounter.
- Any additional data type not defined in this document must not be stored in the same file. Any file containing such additional data type(s) must however always include an Overview.
- Any card download from a given slot must be stored in a separate physical file by the company, without any Overview. Data stored include only the transferred data and exclude SID, TREP, blockSequence Counter and wrapAroundCounter.
- The stored data file name extension to be used varies in function of the back-office software provider and/or the Member State where the company is located. Currently known values are:
    - .tgd (required by Spain for data files from VU and cards,
    - .v1b and c1b (required by France for data files from VU and cards, respectively),
    - .ddd (tachograph manufacturer specific),
    - .esm (tachograph manufacturer specific).

V. APPLICATION LAYER SERVICES AND PROTOCOL

The application layer services used for remote company card authentication and data downloading shall comply with [ISO 16844-6], itself based on [ISO 14229-1].

The DiagnosticOnCAN communication specified in ISO16844-6 shall be used (based on [ISO15765-3]).

To allow the use of the remote download function, the CAN interface defined in section VIII shall be configured by the vehicle manufacturer or by an approved tachograph workshop.

This section only defines additional services and/or sub-functions requirements not mentioned in ISO16844-6.

A specific diagnostic session (remoteSession) must be active in the VU to allow remote company card authentication and data downloading. The DiagnosticSessionControl service shall be used by the FMS to open and close the remoteSession in the VU (DiagnosticSessionType = 0x7E). The accessible services in the remoteSession are defined by the table below:

Diagnostic Service name	Service Id (hex value)
Diagnostic and communication management functional unit	
`DiagnosticSessionControl`	`10`
`TesterPresent`	`3E`
Remote activation of routine functional unit	
`RoutineControl`	`31`
Upload download functional unit	
`RequestUpload`	`35`
`TransferData`	`36`
`RequestTransferExit`	`37`

In the remoteSession:
- the RoutineControl service shall be used for the remote company card authentication and required data types definition,
- the RequestUpload, TransferData, and RequestTransferExit services shall be used for data downloading from the VU to the FMS,

The use of the 0xFB address for any FMS device is strongly recommended.
Please note that in the case the VU does not respond at all to the FMS requests, it may be required to configure the authorised FMS address in the VU, to match the one actually used by the FMS. This needs to be done by an authorised tachograph workshop.

Session/application layer timeouts: 5000 ms (according timeout S3server, [ISO 15765-3], chapter “Session layer timing parameter definitions”)

V.1 REMOTE COMPANY CARD AUTHENTICATION

V.1.1 RoutineControl RemoteTachographCardDataTransfer service description

The RoutineControl service shall be used for each step of the authentication process, with the routineIdentifier set to 0x0180, meaning RemoteTachographCardDataTransfer.

The RoutineControl request messages shall be used to transmit data from the remote company to the VU, using the following format:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x31**							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5 - n	routineControlOptionRecord[]							

The user optional routineControlOptionRecord (251 bytes max) is defined by this document (see section V.1.2.)

The RoutineControl positive response messages shall be used to transmit data from the VU to the remote company card.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	SID : **0x71** (**0x31**+**0x40**)							
1	routineControlType : startRoutine : **0x01**							
2	routineIdentifier MSB : **0x01**							
3	routineIdentifier LSB : **0x80**							
4 - n	routineStatusRecord[]							

The user optional routineStatusRecord (251 bytes max) is defined by this document (see section V.1.3.)

Note: Only the startRoutine subfunction is used for this application, the stopRoutine and requestRoutineResults subfunctions are not used.

V.1.2 Remote company to VU data transfer messages (Requests)

RoutineControl request messages shall be used to transmit data from the remote company card to the VU, with subfunction set to 0x01 (StartRoutine) and routineIdentifier set to 0x0180 (RemoteTachographCardDataTransfer).

V.1.2.1 RemoteCompanyCardReady

V.1.2.1.1 Message definition

This message indicates to the VU that a remote company card is inserted in an appropriate card reader and ready to start the mutual authentication protocol. It also contains the remote company card ATR.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x31**							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineControlOption#1 (RemoteCompanyCardReady) : **0x01**							
6 - (6+n-2)	routineControlOption#2-#n : ATR							

After this message has been correctly received, the VU starts the Tauth timeout (see IV.2.1).

V.1.2.1.2 Execution conditions

No valid company card, control card or workshop card is inserted in the VU.
RemoteSession is active.
TAuth is not active (no remote authentication process is in progress and no remote authentication is still valid).

V.1.2.1.3 Error cases

A valid company card, control card or workshop card is inserted in the VU.
RemoteSession is not active.
TAuth is active (another remote authentication process is already in progress, or a remote authentication is still valid).

V.1.2.2 CompanyCardToVUData

This message contains a tachograph card APDU from the remote company card to the VU.

The FMS shall always send the next CompanyCardToVUData request of the authentication process to the VU after having received a previous VuToCompanyCardData positive response from the VU. Exception: the FMS shall send the first CompanyCardToVUData request at the begin of the authentication process, without having received any previous VuToCompanyCardData positive response from the VU (See example in Annex 1, IX.1).

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x31**							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineControlOption#1 (CompanyCardToVUData) : **0x03**							
6 - (6+n-2)	routineControlOption#2-#n : tachograph card APDU remote company card -> VU							

Note: in the case the APDU content (e.g. status word SW1 SW2) received from the card shows an execution error, the VU retries to send the preceding APDU message sent to the card. After 3 consecutive unsuccessful tries, the VU shall send a positive response with RoutineStatusOption = APDUError (see V.1.3.6). This might happen if the remote company card is expired !

V.1.2.2.1 Execution conditions

No valid company card, control card or workshop card is inserted in the VU.
RemoteSession is active.
A remote company card authentication process has been successfully started (via RemoteCompanyCardReady).
Tauth timeout is active.
2*Trem timeout is active.

V.1.2.2.2 Error cases

A valid company card, control card or workshop card is inserted in the VU.
RemoteSession is not active.
No remote company card authentication process has been successfully started before (via RemoteCompanyCardReady).
Tauth timeout has expired.
2*Trem timeout has expired.

V.1.2.3 RemoteDownloadDataRequest

V.1.2.3.1 Message definition

This message contains the data types required by the company.
The required type of transfer may be selected among the 6 data types defined in [Annex1B Appendix 7] and additional types:

- Overview (mandatory for the FMS within any data transfer of activities, events/faults, detailed speed or technical data),
- Activities of a specified time period,
- Events and faults,
- Detailed speed,
- Technical data,
- Card download (in this case, selected driver card that are currently inserted in the VU shall be downloaded),
- Additional data not defined in this document (optional, VU manufacturer specific)

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x31**							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineControlOption#1 (RemoteDownloadDataRequest): **0x07**							
6 – (6+n-2)	routineControlOption#(2 – n) : DownloadRequestList[]							

The DownloadRequestList is specified as:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	DataTransferRequest #1							
2	DataTransferRequest #1 parameter length							
3 - n	DataTransferRequest #1 parameter (C)							
n+1	DataTransferRequest #2 (U)							
n+2	DataTransferRequest #2 parameter length (U)							
n+3 – m	DataTransferRequest #2 parameter (U) (C)							
...	…							
k+1	DataTransferRequest #N (U)							
k+2	DataTransferRequest #N parameter length (U)							
k+3 – p	DataTransferRequest #N parameter (U) (C)							

Notes:

- the DataTransferRequest parameters, indicated with (C) are conditional, they depend on the preceding DataTransferRequest value.
- the DataTransferRequests #2 to #N are user optional and are indicated with (U).
- The routineControlOption value size is limited to 251 bytes max.

For each DataTransferRequest type, the following table defines the type of data to be downloaded, the DataTransferRequest parameter length value and the allowed parameter values.

DataTransfer Request type	Type of data to be downloaded	Parameter length (bytes)	Parameter
0x01	Overview	0x00	None
0x02	Activities of specified calendar day(s)	0xXX	See next table: activities of specified calendar day(s) parameter
0x03	Events and faults	0x00	None
0x04	Detailed speed	0x00	None
0x05	Technical data	0x00	None
0x06	Card download	0x01	Card slot: 0x01: driver slot 0x02: co-driver slot
0x07-0xFF	Not defined in this document (VU manufacturer specific)	0xXX	VU manufacturer specific

The following table defines the parameter values for the activities of specified calendar day(s).

Byte#	Description	Value (see definition below)
1	`Data input type#1`	0xXX
2 - 5	`Calendar day#1`	0xXX XX XX XX
6	`Data input type#2` (U)	0xXX
7 - 10	`Calendar day#2` (U)	0xXX XX XX XX
…	…	…
n-4	`Data input type#N` (U)	0xXX
(n-3) - n	`Calendar day#N` (U)	0xXX XX XX XX

Note: The data input types #2 to #N are user optional and indicated by (U)

Data input type values:
- Specific day: 0x01
- Period start: 0x02 (shall be immediately followed by a period end)
- Period end: 0x03 (shall immediately follow a period start)

Calendar day values:
- TimeReal format as defined in [Annex1B Appendix 11]

After the reception of this request, rights are opened to download the specified data through the same communication link as the one used for the authentication of the remote company card.

Only one DownloadRequestList is allowed during an authentication process. Even if the VU has already positively answered to a previous RemoteDownloadDataRequest, it shall however accept a new RemoteDownloadDataRequest, with the same DownloadRequestList.

The VU shall answer positively to a RemoteDownloadDataRequest, even if some data requested in the DownloadRequestList are not available in the VU.

V.1.2.3.2 Execution conditions

No valid company card, control card or workshop card is inserted in the VU.
RemoteSession is active.
A RemoteAuthenticationSucceeded has been previously sent by the VU.
No different DownloadRequestList has been previously received by the VU in the current authentication process.
Tauth timeout is active.
2*Trem timeout is active.

V.1.2.3.3 Error cases

A valid company card, control card or workshop card is inserted in the VU.
RemoteSession is not active.

No RemoteAuthenticationSucceeded has been previously sent by the VU.
A different DownloadRequestList has already been received by the VU in the current authentication process.
Tauth timeout has expired.
2*Trem timeout has expired.

V.1.2.4 CloseRemoteAuthentication

V.1.2.4.1 Message definition

This message indicates that the Back-Office PC ends / terminates the authentication process. This can be caused by e.g. the communication problems between the Back Office PC and the remote company card or a break by the user.

This message is also used by the company / FMS to enforce to close a valid remote authentication through this channel. After having received a positive response for this message a new remote authentication process might start.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x31**							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineControlOption#1 (CloseRemoteAuthentication): **0x09**							

The Tauth timeout is closed by the VU after this message has been correctly received (see IV.2.1).

V.1.2.4.2 Execution conditions

RemoteSession is active.

V.1.2.4.3 Error cases

RemoteSession is not active.

V.1.3 VU to remote company card data transfer messages (Positive responses)

RoutineControl positive response messages shall be used to transmit data from the VU to the remote company card, with subfunction set to 0x01 (StartRoutine) and routineIdentifier set to 0x0180 (RemoteTachographCardDataTransfer).

V.1.3.1 VUReady

The VU confirms its readiness for the remote authentication process.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x71** (**0x31**+**0x40**)							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineStatus#1 (VUReady): **0x02**							

V.1.3.2 VUToCompanyCardData

This positive response contains a tachograph card APDU from the VU to the remote company card.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x71** (**0x31**+**0x40**)							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineStatus#1 (VUToCompanyCardData): **0x04**							
6 – (6+n-2)	routineStatus#2 - n : tachograph card APDU VU -> remote company card							

V.1.3.3 RemoteAuthenticationSucceeded

The VU signals that the remote mutual authentication process has succeeded.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x71** (**0x31**+**0x40**)							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineStatus#1 (RemoteAuthenticationSucceeded): **0x06**							

V.1.3.4 RemoteDownloadAccessGranted

The VU signals that the download access to the requested data is granted.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x71** (**0x31**+**0x40**)							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineStatus#1 (RemoteDownloadAccessGranted): **0x08**							

V.1.3.5 RemoteAuthenticationClosed

The VU signals that the authentication process is ended / terminated, or a previously valid authentication closed. The Back Office PC may re-start the authentication process after that.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x71** (**0x31**+**0x40**)							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineStatus#1 (RemoteAuthenticationIsClosed): **0x0A**							

V.1.3.6 APDUError

The VU informs the company that 3 consecutive errors have occurred with APDU sent to the card.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x71** (**0x31**+**0x40**)							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineStatus#1 (APDUError): **0x0C**							

When this message is sent by the VU, the current authentication process is cancelled and Tauth closed. The company must be informed of this error; therefore, this message is a positive response to the FMS, and shall be processed as such by the FMS.

V.1.3.7 AuthenticationError

The VU informs the company that the card authentication has failed (e.g. because of invalid card public key, invalid card member state public key, failed card certificate verification, failed card member state certificate verification, card type different from ‘company’, failed card authentication token verification)

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x71** (**0x31**+**0x40**)							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineStatus#1 (AuthenticationError): **0x0E**							

When this message is sent by the VU, the current authentication process is cancelled and Tauth closed. The company must be informed of this error, therefore this message is a positive response to the FMS, and shall be processed as such by the FMS.
Please note that this error message is sent if the remote company card is expired.

V.1.3.8 TooManyAuthenticationErrors

The VU informs the company that 5 consecutive authentication errors have occurred (as requested by [Annex1B Appendix10], UIA_220)

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x71** (**0x31**+**0x40**)							
2	routineControlType : startRoutine : **0x01**							
3	routineIdentifier MSB : **0x01**							
4	routineIdentifier LSB : **0x80**							
5	routineStatus#1 (TooManyAuthenticationErrors): **0x10**							

When this message is sent by the VU, the current authentication process is cancelled and Tauth closed. The company must be informed of this error; therefore, this message is a positive response to the FMS, and shall be processed as such by the FMS.
Please note that this error message is sent if the remote company card is expired.

V.1.4 Negative responses

In any error case, the VU sends a negative response indicating the error cause.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	NACK : **0x7F**							
2	SID : **0x31**							
3	Negative response codes: - generalReject: **0x10** communication timeout with the remote company card (2 * Trem) has expired. Tauth timeout has expired or has been closed. the VU is not able (for internal reasons) to perform the remote card authentication, the company should try again later. too many errors on low communication layers. a valid company card, control card or workshop card is inserted in the VU. - serviceNotSupported: **0x11** the requested service is not supported. - subFunctionNotSupported: **0x12** the routineControlType parameter is neither startRoutine, stopRoutine nor requestRoutineResults. - incorrectMessageLengthOrInvalidFormat: **0x13** - busyRepeatRequest: **0x21** the VU is busy. The FMS shall perform repetition of this request. - conditionsNotCorrect :**0x22** TAuth is active (another remote authentication process or data transfer is already in progress). a different DownloadRequestList has already been received by the VU in the current authentication process. - requestSequenceError :**0x24** the sequence of the requests of the authentication process is not correct. the stopRoutine or requestRoutineResults subfunction is received, without having first received a startRoutine for the requested routineIdentifier. - requestOutOfRange: **0x31** the routineIdentifier parameter is not supported (e.g. routineIdentifier 0x0180 is used in subfunctions stopRoutine or requestRoutineResults). the optional routineControlOptionRecord is not allowed in or contains invalid data for the requested routineIdentifier (e.g. max APDU size read in RemoteCompanyCardReady is strictly below 240 bytes or strictly above 250 bytes, or a period start in the activities of specified calendar day(s) parameter in a RemoteDownloadDataRequest is not immediately followed by a period stop…). - requestCorrectlyReceived_ResponsePending : **0x78** the request is received well and allowed, but the VU needs more time and "ResponsePending" Messages will be send by the VU until final "PositiveResponse" or "NegativeResponse". - serviceNotSupportedInActiveSession : **0x7F** the current session does not support the StartRoutine (RemoteTachographCardDataTransfer) service, only allowed in remoteSession.							

V.2 DATA DOWNLOADING

V.2.1 RequestUpload

V.2.1.1 Service description

The RequestUpload service is used by the FMS to specify to the VU that a remote data transfer is requested.

Note: a data transfer from the VU to the FMS corresponds to a 'data upload' according to [ISO14229-1], because it is seen from the point of view of the FMS, which is acting as the client.

The RequestUpload Request parameters are defined as follows:
- dataFormatIdentifier is set to 0x00 (because the Data transferred data are neither compressed nor encrypted),
- addressAndLengthFormatIdentifier is set to 0x44 (in order to keep the same memorySize and memoryAddress parameters as the ones defined in [Annex1B Appendix 7], i.e.
  - Length (number of bytes) of the memorySize parameter is set to 4 bytes,
  - Length (number of bytes) of the memoryAddress parameter is set to 4 bytes.
- MemoryAddress is set to 0x00000000 (as it is not known to the FMS prior to a data transfer),
- MemorySize is set to 0xFFFFFFFF (as it is not known to the FMS prior to a data transfer),

The RequestUpload Positive Response parameters are defined as follows:
- the lengthFormatIdentifier is set to 0x10, meaning that the length (number of bytes) of the maxNumberOfBlockLength parameter is set to 1 byte,
- the maxNumberOfBlockLength parameter is set to 0xFF, meaning that 255 data bytes are included in each TransferData Positive Response from the VU (including the service identifier).

In any error case, the VU sends a negative response to the RequestUpload Request.

V.2.1.2 Execution conditions

No valid company card, control card or workshop card is inserted in the VU.
RemoteSession is active.
No data transfer is already in progress.
A remote authentication is valid (access to required data has been granted by the VU and Tauth is active).

V.2.1.3 Error cases

A valid company card, control card or workshop card is inserted in the VU.
RemoteSession is not active.
Another data transfer is already in progress.
No remote authentication is valid (no rights to download data are open, or Tauth is not valid).

V.2.1.4 Messages definition

Request:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x35**							
2	dataFormatIdentifier : **0x00**							
3	addressAndLengthFormatIdentifier : **0x44**							
4	MemoryAddress#1 : **0x00** (MSB)							
5	MemoryAddress#2 : **0x00**							
6	MemoryAddress#3 : **0x00**							
7	MemoryAddress#4 : **0x00** (LSB)							
8	MemorySize#1 MSB : **0xFF** (MSB)							
9	MemorySize#2 MSB : **0xFF**							
10	MemorySize#3 MSB : **0xFF**							
11	MemorySize#4 MSB : **0xFF** (LSB)							

Positive response:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID : **0x75** (**0x35**+**0x40**)							
2	lengthFormatIdentifier: **0x10**.							
3	maxNumberOfBlockLength: **0xFF**							

Negative response:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	NACK : **0x7F**							
2	SID : **0x35**							
3	Negative response codes: - generalReject: **0x10** Tauth has expired or has been closed. a valid company card, control card or workshop card is inserted in the VU. - serviceNotSupported: **0x11** the requested service is not supported. - incorrectMessageLengthOrInvalidFormat: **0x13** - busyRepeatRequest: **0x21** the VU is busy. The FMS shall perform repetition of this request. - conditionsNotCorrect :**0x22** a data transfer is already in progress, but not yet completed. rights to download data are not open (no remote download access to data has been granted). - requestOutOfRange: **0x31** the specified dataFormatIdentifier is not equal to 0x00. the specified addressAndLengthFormatIdentifier is not equal to 0x44. the specified memoryAddress is not equal to 0x00000000. the specified memorySize is not equal to 0xFFFFFFFF. - requestCorrectlyReceived_ResponsePending : **0x78** the request is received well and allowed, but the VU needs more time and "ResponsePending" Messages will be send until final "PositiveResponse" or "NegativeResponse". - serviceNotSupportedInActiveSession : **0x7F** the current session does not support the RequestUpload service, only allowed in remoteSession.							

V.2.2 TransferData

V.2.2.1 Service description

V.2.2.1.1 Generals

The TransferData service shall be used by the FMS to transfer data from the VU to the FMS.

The data transfer direction is defined by the preceding RequestUpload service: data is transferred from the VU to the FMS. The data to be transferred is included in the transferResponseParameter parameters in the TransferData responses.

The transferred data is the one stored in the VU at the time the TransferData request is received.

If a driver card is inserted in the VU after a remote company card has been authenticated, any requested card download for this slot shall be accepted by the VU, however it has to be indicated in the data request list during the authentication process.

V.2.2.1.2 TransferData request

The TransferData request shall include:
- the TransferData SID (0x36),
- a 1 byte blockSequenceCounter,
- a transferRequestParameterRecord (TRTPRecord), which specifies the type of data to be transferred. Rights to download the specified data must have been opened during the authentication process (see V.1.2.3).

The transferRequestParameterRecord shall include:
- a 1 byte wrapAroundCounter
- 1 byte (TRTP#2) specifying the type of data to be transferred.
- additional data for specific transfers i.e. date and slot.

The TRTP#2 value is defined below:

TRTP#2 Hex code	Data to be transferred (0x01to 0x06 as defined by Annex1B Appendix 7])
0x01	Overview
0x02	Activities of a specified calendar day, followed by the value of the calendar day to be downloaded on 4 bytes (`TimeReal` format defined in [Annex1B Appendix 1])
0x03	Events and faults
0x04	Detailed speed
0x05	Technical data
0x06	Card download, followed by the value of the required slot to be downloaded (0x01: driver slot, 0x02: co-driver slot)
0x07 to FF	Not defined by this document (VU Manufacturer specific)

The VU shall consider that a new file transmission is required by the FMS after any correct reception of a TransferData request with blockSequenceCounter=0x01 and wrapAroundCounter=0x00.

So, the blockSequenceCounter parameter value starts at 0x01 with each TransferData request for a new file transmission, as indicated by its TRTP#2 byte. Then, its value is incremented by 1 for each subsequent TransferData request. At the value of 0xFF, the blockSequenceCounter rolls over and starts at 0x00 hex with the next TransferData request.

The wrapAroundCounter parameter value starts at 0x00 with each TransferData request for a new file transmission, as indicated by its TRTP#2 byte. Its value is incremented by 1 for each time the blockSequenceCounter rolls over from 0xFF to 0x00. At the value of 0xFF, the wrapAroundCounter rolls over and starts at 0x01 hex.

So, the wrapAroundCounter / blockSequenceCounter shall be sequentially incremented at each correct reception of a TransferData request, and shall be reset when TransferData with wrapAroundCounter= 0x00 and

blockSequenceCounter= 0x01 is requested.

The FMS shall request to upload data of a given file from its beginning, by sending a TransferData request with:
- TRTP#2 byte set to the requested file value,
- wrapAroundCounter equal to 0x00,
- blockSequenceCounter equal to 0x01.

After any interruption of a data transfer, the remoteSession being active in the VU, the FMS shall request to upload data of a given file from the point of interruption, by sending a TransferData request with:
- TRTP#2 byte set to the requested file value,
- wrapAroundCounter equal to its value at the time of interruption,
- blockSequenceCounter equal to its value at the time of interruption.

If a TransferData request to upload data is correctly received and processed in the VU, but the positive response message does not reach the FMS, then the FMS would determine an application layer timeout and would repeat the same request (including the same wrapAroundCounter and blockSequenceCounter). The VU would receive the repeated TransferData request and could determine based on the included wrapAroundCounter and blockSequenceCounter that this TransferData request is repeated. The VU would send the positive response message immediately, accessing the previously provided data once again in its memory.

If the TransferData request to upload data is not received correctly in the VU, then the VU would not send a positive response message. The FMS would determine an application layer timeout and would repeat the same request (including the same wrapAroundCounter and blockSequenceCounter). The VU would receive the repeated TransferData request and could determine based on the included wrapAroundCounter and blockSequenceCounter that this is a new TransferData. The VU would process the service and would send the positive response message.

Please note that it might be not possible to restart downloading at the point of interruption. In this case the requested file has to be reloaded from the beginning.

Please note that the transferred data reflects the content of the VU or card memory at a certain moment during the download process. Because the download process may last up to 24 hours, different data types within the same download process may not reflect exactly the same content of the VU or card memory. Therefore the VU may answer differently to the same FMS request, depending upon the time at which the request has been received by the VU.
Example:
- The overview file is requested by the FMS on July 7th 2008 at 22:35. At that time the downloadable period of the VU is from March 14th 2007 until July 7th 2008 at 21:47.
- On July 7th 2008 at 23:15, the FMS sends a TransferData request of the activities of July 8th 2008 to the VU. Then the VU answers negatively, because at that time there is no stored activity for that calendar day.
- On July 8th 2008 at 06:45, while the same company authentication has been valid, the FMS sends a TransferData request of the activities of July 8th 2008 to the VU. Then the VU may answer positively, because at that time activities could be stored for that calendar day.

V.2.2.1.3 TransferData positive response

The TransferData positive response shall include:
- the TransferData positive response SID (0x76),
- the 1 byte blockSequenceCounter, which is the echo of the blockSequenceCounter parameter from the request message,
- a transferResponseParameterRecord (TREPRecord).

The transferResponseParameterRecord shall include:
- a 1byte wrapAroundCounter, which is the echo of the wrapAroundCounter parameter from the request message,
- a TREP#2 byte, which is 0x21 to 0x25 (corresponding to TRTP#2 (0x01 to 0x05) in the request message) when response data is in generation 2 format. Otherwise it's the echo of the TRTP#2 byte from the request message,
- data to be transferred from the VU to the FMS.

As defined by the preceding RequestUpload service, 255 data bytes maximum shall be included in each TransferData positive response message from the VU (including SID, blockSequenceCounter, wrapAroundCounter and TREP#2). If the last TransferData positive response message contains exactly 255 bytes in the data field, there is no way for the FMS to detect that no more data is available. The FMS will then continue to send the next TransferData request to the VU. The VU shall answer with a final TransferData positive response message with a data field containing only SID, blockSequenceCounter, wrapAroundCounter and TREP#2. This makes it possible for the FMS to detect that no more data is available.

V.2.2.2 Execution conditions

No valid company card, control card or workshop card is inserted in the VU.
RemoteSession is active.
A preceding RequestUpload service was successfully executed.
Rights to download the requested data are open (the required data are included in the ones specified during the authentication process).
Tauth is valid.
In the case of a requested card download (TRTP#2=0x06) : a driver card is inserted in the relevant slot.

V.2.2.3 Error cases

A valid company card, control card or workshop card is inserted in the VU.
RemoteSession is not active.
No preceding RequestUpload service was successfully executed.
No rights to download the requested data are open (the required data are not included in the ones specified during the authentication process).
Tauth is not valid.
A card download is requested, but no driver card is inserted in the relevant slot even if there has been a card during the authentication.

V.2.2.4 Messages definition

Request:

1) TransferData Request for Overview, Events and faults, Detailed speed and Technical data:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID: **0x36**							
2	blockSequenceCounter							
3	transferRequestParameter#1 (wrapAroundCounter)							
4	transferRequestParameter#2, as defined below: **0x01** : TransferData Request Overview **0x03** : TransferData Request Events and faults **0x04** : TransferData Request Detailed speed **0x05** : TransferData Request Technical data							

2) TransferData Request for activities of a specified date:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID: **0x36**							
2	blockSequenceCounter							
3	transferRequestParameter#1 (wrapAroundCounter)							
4	transferRequestParameter#2, as defined below: **0x02** : TransferData Request Activities							
5	transferRequestParameter#3, CalendarDay#1 (MSB)							
6	transferRequestParameter#4, CalendarDay#2							
7	transferRequestParameter#5, CalendarDay#3							
8	transferRequestParameter#6, CalendarDay#4 (LSB)							

3) TransferData Request for Card Download

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID: **0x36**							
2	blockSequenceCounter							
3	transferRequestParameter#1 (wrapAroundCounter)							
4	transferRequestParameter#2, as defined below: **0x06** : TransferData Request Card download							
5	transferRequestParameter#3, SlotNumber as defined below: **0x01** : Driver slot **0x02** : Co-driver slot							

Note: TREP#2 values from 0x07 to 0x20 and from 0x26 to 0xFF are not defined by this document (VU Manufacturer specific).

Positive response:

1) Normal message (containing 255 bytes)

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID: **0x76** (**0x36**+**0x40**)							
2	blockSequenceCounter							
3	transferResponseParameter#1 (wrapAroundCounter)							
4	transferResponseParameter#2 (TREP#2=0x21 to 0x25 (corresponding to TRTP#2 0x01 to 0x05 in the request message) when response data is in generation 2 format. Otherwise echo of the TRTP#2 byte from the request message)							
5	transferResponseParameter#3 (block of transferred data)							
.	.							
.	.							
.	.							
255	transferResponseParameter#253 (block of transferred data)							

2) Last message of a data transfer (containing less than 255 bytes)

When receiving this message, the FMS can determine that it is the last message of the required data transfer.

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID: **0x76** (**0x36**+**0x40**)							
2	blockSequenceCounter							
3	transferResponseParameter#1 (wrapAroundCounter)							
4	transferResponseParameter#2 (TREP#2=0x21 to 0x25 (corresponding to TRTP#2 0x01 to 0x05 in the request message) when response data is in generation 2 format. Otherwise echo of the TRTP#2 byte from the request message)							
5	transferResponseParameter#3 (block of transferred data)							
.	.							
.	.							
.	.							
n (< 255)	transferResponseParameter#n-2 (block of transferred data)							

3) Last message of a data transfer ('empty' message)

When receiving this message, the FMS can determine that it is the last message of the required data transfer (all data having already been received in the previous TransferData positive response messages).

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID: **0x76** (**0x36**+**0x40**)							
2	blockSequenceCounter							
3	transferResponseParameter#1 (wrapAroundCounter)							
4	transferResponseParameter#2 (TREP#2=0x21 to 0x25 (corresponding to TRTP#2 0x01 to 0x05 in the request message) when response data is in generation 2 format. Otherwise echo of the TRTP#2 byte from the request message)							

Negative response:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	NACK: **0x7F**							
2	SID: **0x36**							
3	Negative response codes: - generalReject: **0x10** Tauth has expired or has been closed. a valid company card, control card or workshop card is inserted in the VU. - serviceNotSupported: **0x11** the requested service is not supported. - incorrectMessageLengthOrInvalidFormat: **0x13** - busyRepeatRequest: **0x21** the VU is busy. The FMS shall perform repetition of this request. - conditionsNotCorrect:**0x22** a card download is requested, but no driver card is inserted in the relevant slot. - requestSequenceError:**0x24** no data transfer has been initialised by a preceding RequestUpload service. - requestOutOfRange: **0x31** the TRTP#2 byte of the TransferData request is not supported by the VU. the TRTP#2 byte of the TransferData request is not the one expected by the VU (i.e. corresponding to the TransferData type in progress). the calendar date specified in the TransferData activities request does not correspond to a day for which activities are stored in the VU (e.g. the requested day is not included in the downloadable period found in the related Overview or the VU was not supplied during that day). the slot specified in the TransferData cardDownload request is not valid. the slot specified in the TransferData cardDownload request is not the one expected by the VU (i.e. corresponding to the TransferData type in progress). the access to the requested data is denied because the data is not included within the accepted download request list. - transferDataSuspended: **0x71** the data transfer operation has been halted due to some internal fault in the VU. - wrongBlockSequenceCounter: **0x73** the VU has detected an error in the sequence of the blockSequenceCounter / wrapAroundCounter or cannot send the data corresponding to the requested blockSequenceCounter / wrapAroundCounter values (the repetition of a TransferData request message with a blockSequenceCounter / wrapAroundCounter equal to the one included in the previous TransferData request message shall be accepted by the VU). - requestCorrectlyReceived-ResponsePending **: 0x78** the request is received well and allowed, but the VU needs more time and "ResponsePending" Messages will be send until final "PositiveResponse" or "NegativeResponse". - serviceNotSupportedInActiveSession : **0x7F** the current session does not support the TransferData service, only allowed in remoteSession.							

Please note that the VU from different brands are reacting in a different way if the day requested contains no data or is not available.

V.2.3 RequestTransferExit

V.2.3.1 Service description

This service is used by the FMS to indicate to the VU that the current data transfer between the VU and the FMS is terminated.

The VU sends a RequestTransferExit positive response message to indicate that the data transfer is terminated according to the FMS request.

The VU shall close the rights to download opened during the authentication process, and close any valid authentication. A new successful authentication of a remote company card is necessary before any other remote data download.

V.2.3.2 Execution conditions

No valid company card, control card or workshop card is inserted in the VU.
RemoteSession is active.
A data transfer is in progress (i.e. a RequestUpload request has been positively answered).
Tauth is valid.

V.2.3.3 Error cases

A valid company card, control card or workshop card is inserted in the VU.
RemoteSession is not active.
No data transfer is in progress (i.e. no valid RequestUpload request has been positively answered).
Tauth is not valid.

V.2.3.4 Messages definition

Request:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID: **0x37**							
2	transferRequestParameter#1, as defined below: **0x00** : Terminate DataTransfer Note: In the case the transferRequestParameter is excluded, the VU shall however respond positively to the request.							

Positive response:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	SID: **0x77** (**0x37**+**0x40**)							
2	TransferResponseParameter#1 (TREP#2=echo of the first byte of the TRTP#2 from the request message, if present).							

Negative response:

Byte	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	NACK: **0x7F**							
2	SID: **0x37**							
3	Negative response codes: - generalReject: **0x10** Tauth has expired or has been closed. a valid company card, control card or workshop card is inserted in the VU. - serviceNotSupported: **0x11** the requested service is not supported. - incorrectMessageLengthOrInvalidFormat: **0x13** - busyRepeatRequest: **0x21** the VU is busy. The FMS shall perform repetition of this request. - conditionsNotCorrect:**0x22** - requestSequenceError:**0x24** no data transfer has been initialised (no RequestUpload service has positively answered). - requestOutOfRange: **0x31** the transferRequestParameter#1 is not valid (not equal to **0x00**) - requestCorrectlyReceived_ResponsePending**: 0x78** the request is received well and allowed, but the VU needs more time and "ResponsePending" Messages will be send until final "PositiveResponse" or "NegativeResponse". - serviceNotSupportedInActiveSession **: 0x7F** the current session does not support the RequestTransferExit service, only allowed in remoteSession.							

VI. NETWORK LAYER

The network layer is compliant to [ISO16844-6] (based on [ISO15765-2]).

The VU shall use the flow control mechanism defined in [ISO15765-2], which allows the receiver to inform the sender about the receiver's capabilities.

In particular, the VU shall use the SeparationTimeMin (STmin) received from the FMS. The time the VU is waiting between the transmissions of two ConsecutiveFrames protocol data units shall therefore not be lower than STmin.

VII. DATA LINK LAYER

The data link layer is compliant to [ISO16844-6] (protocol based on [ISO16844-4], address coding method based on [ISO15765-2]).

VIII. PHYSICAL LAYER

The access to the remote download of the VU is established via a standardised connector in the vehicle.

Please note that it is not allowed to connect directly to the VU. The interface for remote download belongs to the internal network of the vehicle.

If no standard connector exists in the vehicle, the truck manufacturer instruction of how to connect must be followed

The connector might be on different location depending on the brand and model of the vehicle.

The referring document describing the connector and more details are published at www.fms-standard.com

For additional information please ask the manufacturer or its local dealer.

IX. ANNEX 1 : MESSAGE SEQUENCE CHARTS

This annex provides examples of message sequences between the FMS and the VU.

IX.1 MESSAGE SEQUENCE FOR MUTUAL AUTHENTICATION

IX.1.1 Mutual authentication without writing any data on the remote company card

FMS	Dir	VU	Notes
StartDiagnosticSession(remoteSession) request	-> <-	Positive Response StartDiagnosticSession	
RoutineControl request, RemoteCompanyCardReady (ATR)	-> <-	Positive Response RoutineControl, VUReady	
RoutineControl request, CompanyCardToVUData (NoData)	-> <-	Positive Response RoutineControl, VUToCompanyCardData (APDU)	
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	Positive Response RoutineControl, VUToCompanyCardData (APDU)	
... The CompanyCardToVUData – VUToCompanyCardData exchange is performed until the mutual authentication has succeeded. ...			
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	Positive Response RoutineControl, RemoteAuthenticationSucceeded	
RoutineControl request, RemoteDownloadDataRequest (RequestList)	-> <-	Positive Response RoutineControl, RemoteDownloadAccessGranted	
Data transfer from the VU to the FMS can start.			
The company can start communicating with another vehicle.			

IX.1.2 Mutual authentication with writing data on the remote company card (optional)

FMS	Dir	VU	Notes
StartDiagnosticSession(remoteSession) request	-> <-	Positive Response StartDiagnosticSession	
RoutineControl request, RemoteCompanyCardReady (ATR)	-> <-	Positive Response RoutineControl, VUReady	
RoutineControl request, CompanyCardToVUData (NoData)	-> <-	Positive Response RoutineControl, VUToCompanyCardData (APDU)	
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	Positive Response RoutineControl, VUToCompanyCardData (APDU)	
... The CompanyCardToVUData – VUToCompanyCardData exchange is performed until the mutual authentication has succeeded. ...			
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	Positive Response RoutineControl, RemoteAuthenticationSucceeded	
RoutineControl request, RemoteDownloadDataRequest (RequestList)	-> <-	Positive Response RoutineControl, VUToCompanyCardData (APDU)	
... The CompanyCardToVUData – VUToCompanyCardData exchange is performed until the writing to the card has succeeded. ...			
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	Positive Response RoutineControl, RemoteDownloadAccessGranted	
Data transfer from the VU to the FMS can start.			
The company can start communicating with another vehicle.			

IX.2 MESSAGE SEQUENCE: TIMEOUT DURING MUTUAL AUTHENTICATION

FMS	Dir	VU	Notes
StartDiagnosticSession(remoteSession) request	-> <-	Positive Response StartDiagnosticSession	
RoutineControl request, RemoteCompanyCardReady (ATR)	-> <-	Positive Response RoutineControl, VUReady	
RoutineControl request, CompanyCardToVUData (NoData)	-> <-	Positive Response RoutineControl, VUToCompanyCardData (APDU)	
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	Positive Response RoutineControl, VUToCompanyCardData (APDU)	
TesterPresent request	-> <-	Positive Response TesterPresent	Used to keep the remoteSession active in the VU
			Timeout 2*Trem
RoutineControl request, RemoteCompanyCardReady (ATR)	-> <-	Positive Response RoutineControl, VUReady	
Mutual authentication restarts (according to IX.1.1 or IX.1.2)			

IX.3 MESSAGE SEQUENCE: INCORRECT APDU RECEIVED DURING MUTUAL AUTHENTICATION

FMS	Dir	VU	Notes
Mutual authentication has started (according to IX.1.1 or IX.1.2)			
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	Positive Response RoutineControl, VUToCompanyCardData (APDU)	Error in the APDU received by the VU
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	Positive Response RoutineControl, VUToCompanyCardData (APDU)	Error in the APDU received by the VU
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	Positive Response RoutineControl, APDUError	Error in the APDU received by the VU
Mutual authentication aborted			
RoutineControl request, RemoteCompanyCardReady (ATR)	-> <-	Positive Response RoutineControl, VUReady	
Mutual authentication restarts (according to IX.1.1 or IX.1.2)			

IX.4 MESSAGE SEQUENCE: UNSUCCESSFUL MUTUAL AUTHENTICATION

FMS	Dir	VU	Notes
Mutual authentication has started (according to IX.1.1 or IX.1.2)			
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	 Positive Response RoutineControl, AuthenticationError	Authentication error
Mutual authentication aborted			
RoutineControl request, RemoteCompanyCardReady (ATR)	-> <-	 Positive Response RoutineControl, VUReady	
Mutual authentication restarts (according to IX.1.1 or IX.1.2)			
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	 Positive Response RoutineControl, AuthenticationError	Authentication error
...			
RoutineControl request, CompanyCardToVUData (APDU)	-> <-	 Positive Response RoutineControl, TooManyAuthenticationErrors	5 consecutive authentication errors
Mutual authentication aborted			
RoutineControl request, RemoteCompanyCardReady (ATR)	-> <-	 Positive Response RoutineControl, VUReady	
Mutual authentication restarts (according to IX.1.1 or IX.1.2)			

IX.5 MUTUAL AUTHENTICATION INTERRUPTED BY THE COMPANY

FMS	Dir	VU	Notes
StartDiagnosticSession(remoteSession) request	-> <-	 Positive Response StartDiagnosticSession	
RoutineControl request, RemoteCompanyCardReady (ATR)	->

<- |

Positive Response RoutineControl, VUReady | |
| RoutineControl request, CompanyCardToVUData (NoData) | ->

<- |

Positive Response RoutineControl, VUToCompanyCardData (APDU) | |
| RoutineControl request, CompanyCardToVUData (APDU) | ->

<- |

Positive Response RoutineControl, VUToCompanyCardData (APDU) | |
| ...
The CompanyCardToVUData – VUToCompanyCardData exchange continues.
... | | | |
| RoutineControl request, CloseRemoteAuthentication | ->

<- |

Positive Response RoutineControl, RemoteAuthenticationClosed | The company requests to cancel the current authentication process. |
| StartDiagnosticSession (standardSession) request | ->
<- |
Positive Response StartDiagnosticSession | (or the VU exits the remoteSession by session timeout) |
| ... | | | |
| StartDiagnosticSession(remoteSession) request | ->
<- |
Positive Response StartDiagnosticSession | A new authentication process is starting |
| RoutineControl request, RemoteCompanyCardReady (ATR) | ->

<- |

Positive Response RoutineControl, VUReady | |
| RoutineControl request, CompanyCardToVUData (NoData) | ->

<- |

Positive Response RoutineControl, VUToCompanyCardData (APDU) | |
| ... | | | |

IX.6 DATA DOWNLOAD

FMS	Dir	VU	Notes
StartDiagnosticSession(remoteSession) request	-> <-	Positive Response StartDiagnosticSession	If remote session is not already active in the VU
RequestUpload request	-> <-	Positive Response RequestUpload	
TransferData request (wrapAround/blockSequence counter = 0x00/0x01, Required data type n°1)	-> <-	Positive Response TransferData (wrapAround/blockSequence counter = 0x00/0x01, Data)	First required data type (e.g. overview)
TransferData request (wrapAround/blockSequence counter = 0x00/0x02, Required data type n°1)	-> <-	Positive Response TransferData (wrapAround/blockSequence counter = 0x00/0x02, Data)	
... (until all data of the first required type have been transferred)			
TransferData request (wrapAround/blockSequence counter, Required data type n°1)	-> <-	Positive Response TransferData (wrapAround/blockSequence counter, Data)	Last data block of the first required data type
TransferData request (wrapAround/blockSequence counter = 0x00/0x01, Required data type n°2)	-> <-	Positive Response TransferData (wrapAround/blockSequence counter = 0x00/0x01, Data)	Second required data type
TransferData request (wrapAround/blockSequence counter = 0x00/0x02, Required data type n°2)	-> <-	Positive Response TransferData (wrapAround/blockSequence counter = 0x00/0x02, Data)	
... (until all data of the second required type have been transferred)			
TransferData request (wrapAround/blockSequence counter, Required data type n°2)	-> <-	Positive Response TransferData (wrapAround/blockSequence counter, Data)	Last data block of the second required data type
... (all other required data types are transferred the same way)			
TransferData request (wrapAround/blockSequence counter = 0x00/0x01, last required data type)	-> <-	Positive Response TransferData (wrapAround/blockSequence counter = 0x00/0x01, Data)	Last required data type
TransferData request (wrapAround/blockSequence counter = 0x00/0x02, last required data type)	-> <-	Positive Response TransferData (wrapAround/blockSequence counter = 0x00/0x02, Data)	
... (until all data of the last required type have been transferred)			
TransferData request (wrapAround/blockSequence counter, last required data type)	-> <-	Positive Response TransferData (wrapAround/blockSequence counter, Data)	Last data block of the last data type

FMS	Dir	VU	Notes
RequestTranferExit request	-> <-	Positive Response RequestTranferExit	Any still valid remote company authentication is closed.

X. ANNEX 2: USER GUIDANCE FOR ERROR CASES

The following tables provide user guidance for processing any error case during the remote authentication and data download process.

Each table applies for a different step:

- Remote authentication,
  - Table 1: Negative response codes to a RoutineControl request)
  - Table 2: Positive response messages to a RoutineControl request, corresponding to error cases

- Data download,
  - Table 3: Negative response codes to a RequestUpload request
  - Table 4: Negative response codes to a TransferData request
  - Table 5: Negative response codes to a RequestTransferExit request

The 1st column shows the negative response code sent by the VU (NACK, SID, Negative Response code in hexadecimal) or the positive response message content
The 2nd column provides the description of the negative response code or the positive response message content
The 3rd column provides the VU authentication status after the error has occurred
The 4th column lists the possible causes of the error
The 5th column recommends the FMS/Company behaviour to adopt in this case

Some general annotations:

- It is recommended to send a Close Authentication before starting a remote Authentication process.
- If a valid Company Card, Workshop Card or Control Card is inserted in the VU a remote Authentication is not possible.
- The TransferExit command is used to finish a successful remote downloading and to close the remote Authentication. However, if this command has a negative response it has no influence to the already downloaded files. All data are downloaded correctly (if no error response during downloading) and are valid. However, if the TransferExit command fails the VU is still in Authentication mode (Therefore send Close Authentication before starting a remote Authentication).
- If an error message occurs for more than 24 hours of operation of the tachograph, please check all used components for remote download
- The data downloaded are “frozen” in the moment of a TransferData command (after successful Authentication and successful RequestUpload)
- The last download date and time indicated in the Overview file does not indicate that the data have been successfully received by the company.

Table 1: Negative Response codes during Remote Authentication (RoutineControl request)

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x7F 31 10	generalReject	Remote Authentication not valid TAuth not valid	Communication timeout with the remote company card (2 * Trem) has expired	- System should send Close Authentication - System should start Remote Authentication from the beginning - If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			A valid company card, control card or workshop card is inserted in the VU	- System should send Close Authentication - System should wait ca. 10 minutes - System should start Remote Authentication from the beginning - If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			Tauth timeout has expired Tauth has been closed	- System should send Close Authentication - System should start Remote Authentication from the beginning - If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The VU is not able (for internal reasons) to perform the remote card authentication	- System should send Close Authentication - System should start Remote Authentication from the beginning - If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			Too many errors on low communication layers	- Check the application for correct implementation of the communication protocol - If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 31 11	serviceNotSupported	Remote Authentication not valid TAuth valid	The requested service is not supported by the VU	- Check the application for correct implementation of the communication protocol - System should send Close Authentication - System should start Remote Authentication from the beginning - If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The remote download function is not implemented in the VU	Check whether the VU is supporting remote download function

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x7F 31 12	subFunction NotSupported	Remote Authentication not valid TAuth valid	The routineControlType parameter is neither startRoutine, stopRoutine nor requestRoutineResults	- Check the application for correct implementation of the communication protocol - System should send Close Authentication - System should start Remote Authentication from the beginning - If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 31 13	incorrectMessageLength OrInvalidFormat	Remote Authentication not valid TAuth valid	The request was incorrect	- Check the application for correct implementation of the communication protocol - System should send Close Authentication - System should start Remote Authentication from the beginning - If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The request was incorrectly processed by the VU	- System should send Close Authentication - System should start Remote Authentication from the beginning - If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 31 21	busyRepeatRequest	Remote Authentication not valid TAuth valid	The VU is busy - There are higher priority tasks in the VU than the remote authentication (e.g. card insertion, printout) - The remote authentication process must be delayed for any reason	System should - wait ca. 5 minutes - open the remoteSession again - send the same request again For repeating the complete Remote Authentication, the current Authentication has to be closed before (Close Authentication) If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 31 22	conditionsNotCorrect	Remote Authentication not valid (a different one is active) TAuth (a different one) is valid	TAuth is active (another remote authentication process or data transfer is already in progress)	There is a conflict between several authentication processes addressing the same VU. Check the application for correct implementation (System should prevent access to VU from different remote locations) To force a new remote Authentication: - System should send Close Authentication - System should start Remote Authentication from the beginning
			A different DownloadRequestList has already been received by the VU in the current authentication process.	Check the application for correct implementation (DownloadRequestList)

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x7F 31 24	requestSequenceError	Remote Authentication not valid TAuth valid	The sequence of the requests of the authentication process is not correct.	Check the application for correct implementation - System should send Close Authentication - System should start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The stopRoutine or requestRoutineResults subfunction is received, without having first received a startRoutine for the requested routineIdentifier	Check the application for correct implementation - System should send Close Authentication - System should start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The steps of the authentication process are not executed in the right order in the VU	- System should send Close Authentication - System should start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 31 31	requestOutOfRange	Remote Authentication not valid TAuth valid	The routineIdentifier parameter is not supported (e.g. routineIdentifier 0x0180 is used in subfunctions stopRoutine or requestRoutineResults).	Check the application for correct implementation - System should send Close Authentication - System should start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The optional routineControlOptionRecord is not allowed in or contains invalid data for the requested routineIdentifier (e.g. max APDU size read in RemoteCompanyCardReady is strictly below 240 bytes or strictly above 250 bytes, or a period start in the activities of specified calendar day(s) parameter in a RemoteDownloadDataRequest is not immediately followed by a period stop	Check the application for correct implementation - System should send Close Authentication - System should start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
			The request was incorrect (e.g. wrong card used, wrong data request list format)	Check the application for correct implementation (data request list format, etc.) Check the used cards - System should send Close Authentication - System should start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The request was incorrectly processed by the VU	- System should send Close Authentication - System should start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 31 78	requestCorrectlyReceived_ResponsePending	Remote Authentication not valid TAuth valid	The request is received well and allowed, but the VU needs more time and "ResponsePending" Messages will be send by the VU (e.g. each 4s), until final "PositiveResponse" or "NegativeResponse"	System should wait until the final "PositiveResponse" or "NegativeResponse" is received. There is no possibility to stop this process by the System

0x7F 31 7F	serviceNotSupported InActiveSession	Remote Authentication not valid TAuth valid	The current session does not support the StartRoutine (RemoteTachographCardDataTransfer) service, only allowed in remoteSession	Check the application for correct implementation System should - open remoteSession - continue Remote Authentication (repeat request to VU) If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The VU has exit the remoteSession for any reason (e.g. the FMS has not sent any request to the VU during 5s or more)	Check the application for correct implementation System should - open remoteSession - continue Remote Authentication (repeat request to VU) If this error code is received for more than 24 hours the correct operation of all system parts must be checked

Table 2: Positive Response codes during Remote Authentication (RoutineControl request)

Positive Response Code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x71 01 01 80 0C	The VU informs that 3 consecutive errors have occurred with APDU sent to the card.	Remote Authentication not valid TAuth not valid	Company card read/write problem	- System should send Close Authentication - Check the card (e.g. clean card) - System should start Remote Authentication from the beginning - If the same response occurs, please replace the company card with a new company card and start Remote Authentication again
			Card reader problem	- System should send Close Authentication - Check the card reader (e.g. clean card reader) - System should start Remote Authentication from the beginning - If the same response occurs, please replace the card reader with a new card reader and start Remote Authentication again
			Card damaged	- System should send Close Authentication - Replace the company card with a new company card and start Remote Authentication again
			Card reader use different T=0/T=1 than the VU is requesting	Check the application for correct implementation of the communication protocol with the card reader (see Annex 3)
0x71 01 01 80 0E	The VU informs that the card authentication has failed	Remote Authentication not valid TAuth not valid	Company Card expired	- System should send Close Authentication - Replace the expired company card with a new valid company card and start Remote Authentication again Expired, failed or corrupted company cards must be sent back to the relevant Member State Authority.
			Bad card type (not a Company Card)	- System should send Close Authentication - Replace the bad card with a valid company card and start Remote Authentication again

Positive Response Code	Description	Status VU	Possible cause(s)	Recommended system behaviour
			Company Card corrupted (e.g. invalid card public key, invalid card member state public key, failed card certificate verification, failed card member state certificate verification, card type different from 'company', failed card authentication token verification	- System should send Close Authentication - Replace the bad card with a valid company card and start Remote Authentication again Expired, failed or corrupted company cards must be sent back to the relevant Member State Authority.
			Card reader use different T=0/T=1 than the VU is requesting	Check the application for correct implementation of the communication protocol with the card reader (see Annex 3)
0x71 01 01 80 10	The VU informs that 5 consecutive authentication errors have occurred (as requested by [Annex1B Appendix10], UIA_220)	Remote Authentication not valid TAuth not valid	Invalid card repetitively used	- System should send Close Authentication - Replace the invalid card with a valid company card and start Remote Authentication again Expired, failed or corrupted company cards must be sent back to the relevant Member State Authority.
			Card reader use different T=0/T=1 than the VU is requesting	Check the application for correct implementation of the communication protocol with the card reader (see Annex 3)

Table 3: Negative Response codes for RequestUpload

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x7F 35 10	generalReject	Remote Authentication Not valid TAuth not valid	Tauth timeout has expired or has been closed	The System should - stop the download process - send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			A valid company card, control card or workshop card is inserted in the VU	The System should - stop the download process - send Close Authentication - wait for ca. 10 minutes - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 35 11	serviceNotSupported	Remote Authentication is valid TAuth is valid	The requested service is not supported by the VU	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 35 13	incorrectMessageLength OrInvalidFormat	Remote Authentication is valid TAuth is valid	The request was incorrect	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
			The request was incorrectly processed by the VU	The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 35 21	busyRepeatRequest	Remote Authentication is valid TAuth is valid	The VU is busy - There are higher priority tasks in the VU than the remote authentication (e.g. card insertion, printout) - The remote authentication process must be delayed for any reason	System should - wait ca. 5 minutes - open the remoteSession again - send the same request again If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 35 22	conditionsNotCorrect	Remote Authentication is valid TAuth is valid	A data transfer is already in progress, but not yet completed.	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			Rights to download data are not open (no remote download access to data has been granted)	Check the application for correct implementation
			Bad management in the FMS of simultaneous download requests from several companies	Check the application for correct implementation (System should prevent access to VU from different remote locations) To force a new remote Authentication: - System should send Close Authentication - System should start Remote Authentication from the beginning

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x7F 35 31	requestOutOfRange	Remote Authentication is valid TAuth is valid	The specified dataFormatIdentifier is not equal to 0x00.	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The specified addressAndLengthFormatIdentifier is not equal to 0x44.	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The specified memoryAddress is not equal to 0x00000000.	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The specified memorySize is not equal to 0xFFFFFFFF	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			Bad management in the FMS of simultaneous download requests from several companies	Check the application for correct implementation (System should prevent access to VU from different remote locations) To force a new remote Authentication: - System should send Close Authentication

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
				- System should start Remote Authentication from the beginning
0x7F 35 78	requestCorrectlyReceived_ResponsePending	Remote Authentication is valid TAuth is valid	The request is received well and allowed, but the VU needs more time and "ResponsePending" Messages will be send by the VU (e.g. each 4s), until final "PositiveResponse" or "NegativeResponse"	System should wait until the final "PositiveResponse" or "NegativeResponse" is received. There is no possibility to stop this process by the System
0x7F 35 7F	serviceNotSupportedInActiveSession	Remote Authentication is valid TAuth is valid	The current session does not support the RequestUpload service, only allowed in remoteSession.	Check the application for correct implementation System should - open remoteSession - repeat request to VU
			The VU has exit the remoteSession for any reason (e.g. the FMS has not sent any request to the VU during 5s or more	Check the application for correct implementation System should - open remoteSession - repeat request to VU

Table 4: Negative Response codes during TransferData

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x7F 36 10	generalReject	Remote Authentication not valid TAuth not valid	Tauth timeout has expired or has been closed	The System should - stop the download process - send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			A valid company card, control card or workshop card is inserted in the VU	The System should - stop the download process - send Close Authentication - wait for ca. 10 minutes - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 36 11	serviceNotSupported	Remote Authentication is valid TAuth is valid	The requested service is not supported by the VU	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 36 13	incorrectMessageLengthOrInvalidFormat	Remote Authentication is valid TAuth is valid	The request was incorrect	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
			The request was incorrectly processed by the VU	The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 36 21	busyRepeatRequest	Remote Authentication is valid TAuth is valid	The VU is busy - There are higher priority tasks in the VU than the remote authentication (e.g. card insertion, printout) - The remote authentication process must be delayed for any reason	System should - wait ca. 5 minutes - open the remoteSession again - send the same request again For repeating the complete Remote Authentication, the current Authentication has to be closed before (Close Authentication) If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 36 22	conditionsNotCorrect	Remote Authentication is valid TAuth is valid	A card download is requested, but no driver card is inserted in the relevant slot.	Don't care, continue the current download process Just indicating that there is no Driver Card in the relevant slot
			The relevant driver card has been removed.	Don't care, continue the current download process Just indicating that there is no Driver Card in the relevant slot
0x7F 36 24	requestSequenceError	Remote Authentication is valid TAuth is valid	No data transfer has been initialised by a preceding RequestUpload service.	Check the application for correct implementation The System should - send RequestUpload - send TransferData of all data types specified in the DownloadRequestList
			Bad management of requests sequences	Check the application for correct implementation
0x7F 36 31	requestOutOfRange	Remote Authentication is valid TAuth is valid	The TRTP#2 byte of the TransferData request is not supported by the VU	Check the application for correct implementation - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
			The TRTP#2 byte of the TransferData request is not the one expected by the VU (i.e. corresponding to the TransferData type in progress)	Check the application for correct implementation - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The calendar date specified in the TransferData activities request does not correspond to a day for which activities are stored in the VU (e.g. the requested day is not included in the downloadable period found in the related Overview or the VU was not supplied during that day).	Don't care, continue the current download process Just indicating that there is no data for the day stored in the VU
			The slot specified in the TransferData cardDownload request is not valid	Check the application for correct implementation
			The slot specified in the TransferData cardDownload request is not the one expected by the VU (i.e. corresponding to the TransferData type in progress)	Check the application for correct implementation
			The access to the requested data is denied because the data is not included within the accepted download request list	Check the application for correct implementation continue the current download process Just indicating that this data is not requested and granted from the VU
			The company requested to download activities for a calendar day for which no activities are stored in the VU	Don't care, continue the current download process Just indicating that there is no data for the day stored in the VU
			Bad management of requests sequences	Check the application for correct implementation

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x7F 36 71	transferDataSuspended	Remote Authentication is valid TAuth is valid	the data transfer operation has been halted due to some internal fault in the VU	Check the application for correct implementation - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication - start Remote Authentication from the beginning If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 36 73	wrongBlockSequence Counter	Remote Authentication is valid TAuth is valid	The VU has detected an error in the sequence of the blockSequenceCounter/wrapAround Counter	Check the application for correct implementation The System should retry to download the data type for which the error occurred by requesting the first block of the file from the beginning (TransferData request with TRTP#2 byte set to the requested file value, wrapAroundCounter equal to 0x00, blockSequenceCounter equal to 0x01)
			The VU cannot send the data corresponding to the requested blockSequenceCounter / wrapAroundCounter values (the repetition of a TransferData request message with a blockSequenceCounter / wrapAroundCounter equal to the one included in the previous TransferData request message shall be accepted by the VU	Check the application for correct implementation The System should retry to download the data type for which the error occurred by requesting the first block of the file from the beginning (TransferData request with TRTP#2 byte set to the requested file value, wrapAroundCounter equal to 0x00, blockSequenceCounter equal to 0x01)
			There is a mismatch between the block number expected by the VU and the one requested by the FMS, due to an interruption of the data download	The System should retry to download the data type for which the error occurred by requesting the first block of the file from the beginning (TransferData request with TRTP#2 byte set to the requested file value, wrapAroundCounter equal to 0x00, blockSequenceCounter equal to 0x01)
0x7F 36 78	requestCorrectly Received_ ResponsePending	Remote Authentication is valid TAuth is valid	The request is received well and allowed, but the VU needs more time and “ResponsePending” Messages will be send by the VU (e.g. each 4s), until final “PositiveResponse” or “NegativeResponse”	System should wait until the final “PositiveResponse” or “NegativeResponse” is received. There is no possibility to stop this process by the System

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x7F 36 7F	serviceNotSupported InActiveSession	Remote Authentication is valid TAuth is valid	The current session does not support the TransferData service, only allowed in remoteSession.	Check the application for correct implementation System should - open remoteSession - repeat request to VU: retry to download the data type for which the error occurred by requesting the first block of the file from the beginning (TransferData request with TRTP#2 byte set to the requested file value, wrapAroundCounter equal to 0x00, blockSequenceCounter equal to 0x01)
			The VU has exit the remoteSession for any reason (e.g. the FMS has not sent any request to the VU during 5s or more)	Check the application for correct implementation System should - open remoteSession - repeat request to VU: retry to download the data type for which the error occurred by requesting the first block of the file from the beginning (TransferData request with TRTP#2 byte set to the requested file value, wrapAroundCounter equal to 0x00, blockSequenceCounter equal to 0x01)

Table 5: Negative Response codes during RequestTransferExit

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x7F 37 10	generalReject	Remote Authentication not valid TAuth not valid	Tauth timeout has expired or has been closed	No action (the RequestTransferExit service is used by the System to close the Tauth timeout). This error has no influence on the already downloaded files
			A valid company card, control card or workshop card is inserted in the VU	No action (the RequestTransferExit service is used by the System to close the Tauth timeout). This error has no influence on the already downloaded files
0x7F 37 11	serviceNotSupported	Remote Authentication is valid TAuth is valid	The requested service is not supported by the VU	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication This error has no influence on the already downloaded files If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 37 13	incorrectMessage LengthOr InvalidFormat	Remote Authentication is valid TAuth is valid	The request was incorrect	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication This error has no influence on the already downloaded files If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			The request was incorrectly processed by the VU	The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication This error has no influence on the already downloaded files If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 37 21	busyRepeatRequest	Remote Authentication is valid	The VU is busy - There are higher priority tasks in the VU than the remote authentication (e.g. card insertion, printout)	System should - wait ca. 5 minutes - open the remoteSession again - send the same request again

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
		TAuth is valid	- The remote authentication process must be delayed for any reason	This error has no influence on the already downloaded files If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 37 22	conditionsNot Correct	Remote Authentication is valid TAuth is valid	Unclear: internal conditions in the VU not met	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication This error has no influence on the already downloaded files If this error code is received for more than 24 hours the correct operation of all system parts must be checked
0x7F 37 24	requestSequence Error	Remote Authentication is valid TAuth is valid	no data transfer has been initialised (no RequestUpload service has been positively answered)	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession restart the download process from the beginning (RequestUpload, and then TransferData of all data types specified in the DownloadRequestList) If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			Bad management of requests sequences	Check the application for correct implementation
0x7F 37 31	requestOutOfRange	Remote Authentication is valid TAuth is valid	The transferRequestParameter#1 is not valid (not equal to 0x00)	Check the application for correct implementation The System should - repeat the request - if the error code is received for more than 5 times during the same remoteSession send Close Authentication This error has no influence on the already downloaded files If this error code is received for more than 24 hours the correct operation of all system parts must be checked
			Bad management of requests sequences	Check the application for correct implementation
0x7F 37 78	requestCorrectlyReceived_ResponsePending	Remote Authentication is valid TAuth is valid	The request is received well and allowed, but the VU needs more time and "ResponsePending" Messages will be send by the VU (e.g. each 4s), until final "PositiveResponse" or "NegativeResponse"	System should wait until the final "PositiveResponse" or "NegativeResponse" is received. There is no possibility to stop this process by the System

Negative Response code	Description	Status VU	Possible cause(s)	Recommended system behaviour
0x7F 37 7F	serviceNotSupported InActiveSession	Remote Authentication is valid TAuth is valid	The current session does not support the RequestTransferExit service, only allowed in remoteSession	Check the application for correct implementation System should - open remoteSession - repeat request to VU retry to download the data type for which the error occurred by requesting the first block of the file from the beginning (TransferData request with TRTP#2 byte set to the requested file value, wrapAroundCounter equal to 0x00, blockSequenceCounter equal to 0x01)
			The VU has exit the remoteSession for any reason (e.g. the FMS has not sent any request to the VU during 5s or more)	Check the application for correct implementation System should - open remoteSession - repeat request to VU (retry to download the data type for which the error occurred by requesting the first block of the file from the beginning (TransferData request with TRTP#2 byte set to the requested file value, wrapAroundCounter equal to 0x00, blockSequenceCounter equal to 0x01)

XI. ANNEX 3 : USER GUIDANCE FOR MANAGING THE COMPANY CARD READER

Some card readers cannot automatically react on the requested protocol T0 or T1 from the VU. They chose the protocol on the used card.
Therefore the application in the Fleet Back Office System has to react on the requested protocol by the VU.

Please note that a positive remote authentication is only possible when using the correct protocol !!

Here is an example how to detect and to change the protocol (T0/T1) independent from the used card reader according the requested protocol from the VU.

```
// LogiCom GmbH - Example for support of T=0 T=1 protocol
// Commands: Internal Authentication,Secure Read,Secure Update (Dtco T=0)
// Date: 2008/06/10
// Please do not distribute
//
// Globales
int CardProtocol=0;
int DtcoProtocol;
U08 T0T1Buffer[256];
int T0T1Len;

int  TCardReader_Read( U08* Comand, U16 Len,U08* Answer,U16* LenMax )
{
 int Res;
 U08 Sad=2;
 U08 Dad=0;
 if ( Comand[0]==0x20 )
   Dad=1;

 // Messages without protocoll differences
 if ( Comand[1]!=0x88 && Comand[1]!=0xC0 && Comand[1]!=0xD6 && Comand[0]!=0x0C )
 {
  Res = CT_data ( 1,&Dad,&Sad,Len,Comand,LenMax,Answer );
  return Res;
 }

 // Protocoll detection Internal Authentication
 if ( Comand[0]==0x00 && Comand[1]==0x88 )
 {
  if ( Len == 22 )
    DtcoProtocol = 1;
  if ( Len == 21 )
    DtcoProtocol = 0;
 }

 // Protocoll detection Read secure messaging
 if ( Comand[0]==0x0C && Comand[1]==0xB0 )
 {
  if ( Len == 15 )
    DtcoProtocol = 1;
  if ( Len == 14 )
    DtcoProtocol = 0;
 }
```

```
// Protocoll detection Update secure messaging
if ( Comand[0]==0x0C && Comand[1]==0xD6 )
{
 if ( (Comand[6]+13) == Len )
   DtcoProtocol = 0;
 else
   DtcoProtocol = 1;
}


// No protocoll differences
if ( CardProtocol==DtcoProtocol )
{
  Res = CT_data ( 1,&Dad,&Sad,Len,Comand,LenMax,Answer );
  return Res;
}

// Dtco= T=0 and Cardreader= T=1
if ( (DtcoProtocol==0) && (CardProtocol==1) )
{
  if ( Comand[1]==0xC0 )
  {
    int Res = CT_data ( 1,&Dad,&Sad,T0T1Len,T0T1Buffer,LenMax,Answer );
    return Res;
  }
  memcpy ( T0T1Buffer,Comand,Len);
  T0T1Len = Len+1;
  Answer[0]=0x61;
  *LenMax = 2;
  if ( Comand[1]==0x88 )
  {
    T0T1Buffer[Len]=0x80;
    Answer[1]=0x80;
    return 0;
  }

  if ( Comand[1]==0xB0 )
  {
    T0T1Buffer[Len]=0x00;
    Answer[1]=Comand[7];
    return 0;
  }

  if ( Comand[1]==0xD6 )
  {
    T0T1Buffer[Len]=0x00;
    Answer[1] = 0x0A;
    return 0;
  }
}

// Dtco= T=1 and Cardreader= T=0
if ( (DtcoProtocol==1) && (CardProtocol==0) )
{
  if ( Comand[1]==0x88 )
  {
    Res = CT_data ( 1,&Dad,&Sad,21,Comand,LenMax,Answer );
    if ( *LenMax!=2 )
    {
      return 0;
```

```
      }
    }

    if ( Comand[1]==0xB0 )
    {
      Res = CT_data ( 1,&Dad,&Sad,14,Comand,LenMax,Answer );
      if ( *LenMax!=2 )
      {
        return 0;
      }
    }

    memcpy ( Comand,(U08*)"\x00\xC0\x00\x00",4);
    Comand[4] = Answer[1];

    *LenMax = 255;
    Sad=2;Dad=0;
    Res = CT_data ( 1,&Dad,&Sad,5,Comand,LenMax,Answer );
    return Res;
  }
  return -1;
}
```