GV355CEU TachoReader User Manual

Description of Work

TRACTACHOAN001

Version: 1.00



Driving Smarter IoT

www.queclink.com

Document Title	GV355CEU TachoReader User Manual
Version	1.00
Date	November 25, 2022
Status	Release
Document Control ID	TRACTACHOAN001

General Notes

Queclink offers this information as a service to its customers, to support application and engineering efforts that use the products designed by Queclink. The information provided is based upon requirements specifically provided to Queclink by the customers. Queclink has not undertaken any independent search for additional relevant information, including any information that may be in the customer's possession. Furthermore, system validation of this product designed by Queclink within a larger electronic system remains the responsibility of the customer or the customer's system integrator. All specifications supplied herein are subject to change.

Copyright

This document contains proprietary technical information which is the property of Queclink Wireless Solutions Co., Ltd. The copying of this document, distribution to others, and communication of the contents thereof, are forbidden without express authority. Offenders are liable to the payment of damages. All rights are reserved in the event of a patent grant or registration of a utility model or design. All specification supplied herein are subject to change without notice at any time.

Queclink Confidential

Contents

Contents 2
0. Revision History 3
1. Introduction 4
1.1. Telemetric System 4
1.2. Procedure of DDD Files Remote Download 5
1.3. Requirements 5
1.4. Implementation 6
1.4.1. Request Procedure 6
1.4.2. Sequence Diagram 8

Queclink Confidential

0. Revision History

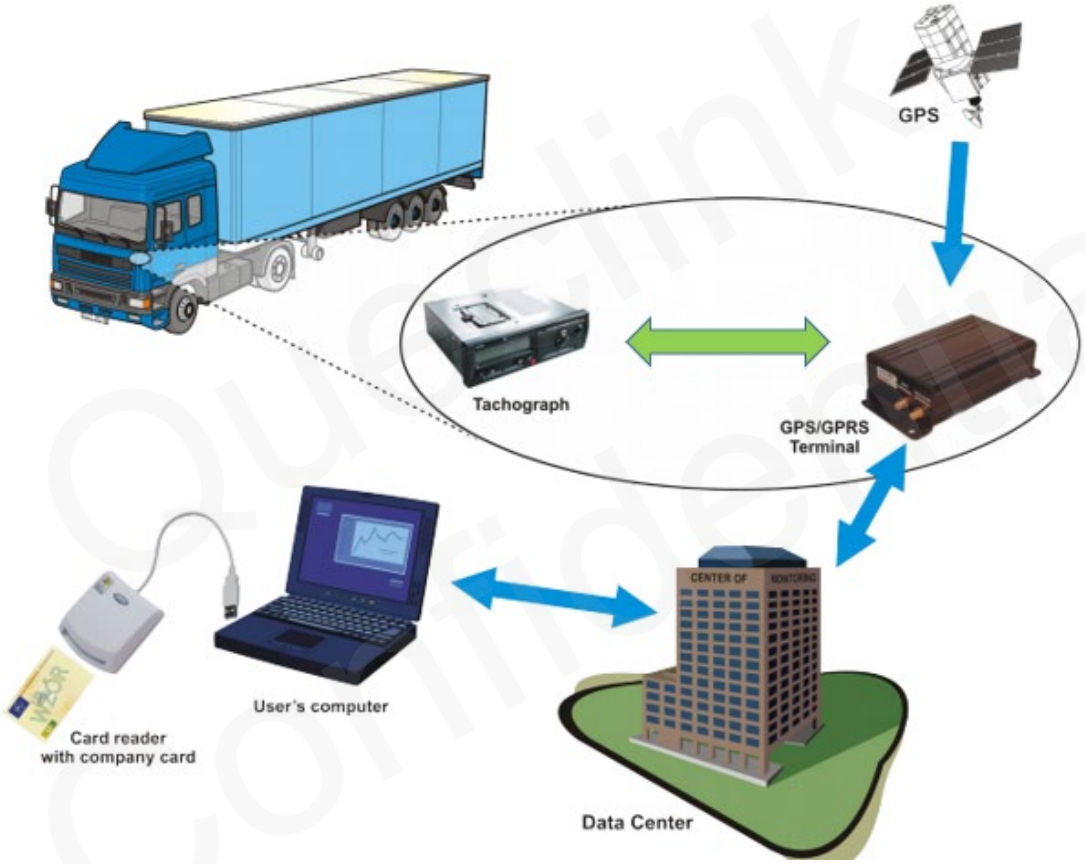
Revision	Date	Author	Description of change
1.00	November 25, 2022	Aleo Liu	Initial.

Queclink Confidential

1. Introduction

1.1. Telemetric System

The terminal can remotely read DDD files from digital tachograph and driver card as a part of telemetric system. Commonly, in order to read files from tachograph, it is necessary to put company card directly into the vehicle's tachograph and download required files with a specific tool. The problem appears when the vehicles are far away in foreign countries but the company card is in the home office.
Data between tachograph and company card is exchanged over the Internet using the existing structure of telemetric system.



Structure of Telemetric System

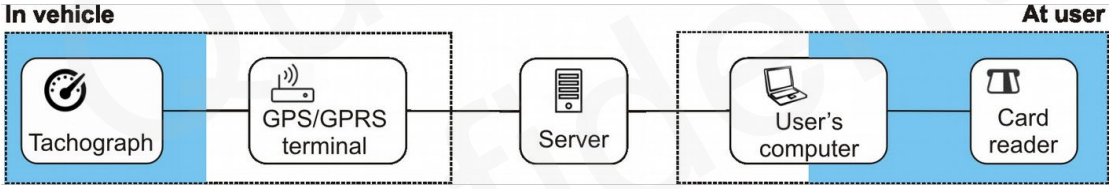
A typical vehicle monitoring system consists of a telemetric device (GNSS/GPRS terminal) placed in a vehicle, a server and software running on end user's computer.
In order to allow a remote download of DDD files from tachograph in a vehicle, the system must be connected with GNSS/GPRS terminal (which is connected to the vehicle's tachograph) and provide its communication with the company card, which is inserted into end user's computer by a card reader.

1.2. Procedure of DDD Files Remote Download

The procedure of reading DDD files from tachograph has 2 stages:
–Authorization, where the tachograph exchanges several encrypted data packets and control commands with the company card.
–Files download, where GNSS/GPRS terminal reads requested data from tachograph memory or driver card.

To perform the authorization, the telemetric system must provide communication between the tachograph and the company card, as shown in the diagram below.
–Card reader with supporting application (TachoCardAuthorizer) to be run on the end user's computer.

Other elements are part of typical telemetric system, that must be adapted by:
–GNSS/GPRS terminal software must be able to send data packets to and from the company card in a timely manner.
–Server software must manage the whole process of remote files download from the time the end user makes a request to providing the DDD files he requested; it must also forward data packets from THR GNSS/GPRS terminal to TachoCardAuthorizer application running on the end user's computer.
–Application running on the backend user's must allow him to make a request of particular DDD files to download from a particular vehicle.



Interactions Between Particular Modules of Telemetric System

Queclink provides elements placed on blue background on the scheme above:
–Communication of GNSS/GPRS terminal with tachograph (all models of tachographs that are available on market and support remote DDD files download), it is required to connect GNSS/GPRS terminal to the tachograph according to the installation manual provided.
–TachoCardAuthorizer application to be run on end user's computer, which provides communication with card reader and company card.
–Communication protocol (over http/https) of TachoCardAuthorizer application with Server.

1.3. Requirements

Communication with GNSS/GPRS terminal requires:
–In GNSS/GPRS Terminal: One free serial port or serial port for communication with

Tachograph.
–Fixed network connection between the Terminal and the Server (typically by GPRS) and between the Server and the end user's computer (at least as long as the procedure of authorization is running).
–Java Runtime Environment (at least version 1.7.0) installed on the end user's computer (requires by TachoCardAuthorizer application).
–One free USB port on the end user's computer to connect card reader.
–Card reader drivers (delivered with the reader).
TachoCardAuthorizer application supports following operation systems (with Java Runtime Environment installed, which can be downloaded free of charge from website: https://www.java.com/en/download/):
–Windows (XP, Vista, 7, 8, 10)
–Linux
–OS X

1.4. Implementation

The implementation of Server is a concern of customer as Queclink provides application TachoCardAuthorizer and GNSS/GPRS terminal device. This chapter contains step-by-step guidelines on how to implement support for GNSS/GPRS terminal on Server.
At the end of this section there is sequence diagram, which shows the entire procedure of downloading DDD files remotely using GNSS/GPRS terminal and card reader application TachoCardAuthorizer.
This procedure refers to:
–TachoCardAuthorizer technical documentation

1.4.1. Request Procedure

Log in
The first step is to configure the TachoCardAuthorizer application to connect to the Server by editing the config.dat file. It is recommended to use debug version of the application and start it from command line to view detailed logs containing all outgoing and incoming http requests.
After start, the application asks the user for login data (username, password), which can be omitted during implementation and tests.

Waiting for files download request (the Request)
Server is the manager of all DDD files download requests, and it has to implement a queue of requests. Each Request has its unique request ID and is related to device with the gpsID number and the user who initiates the request.
TachoCardAuthorizer, after login, queries the Server about queue of Requests available for logged-in user. In response, the Server sends a queue of Requests that are ready to authorize. The queue should be empty until any Request is not initiated and confirmed by the GNSS/GPRS terminal.

Request initiation

User interface and way of its communication with the Server is not a concern of this manual. It is assumed, that there exists an interface, where user can make a Request (selecting the vehicle and the requested files) and enqueue it to the Server.

A new Request with status “queued” appears on Server. Server sends the Request command (including all parameters) to GNSS/GPRS terminal. After GNSS/GPRS terminal accepts the Request, Server can “hand over” it to TachoCardAuthorizer for authorization, place the Request in reply to TachoCardAuthorizer query (GET queue). Application confirms start of authorization procedure for specified Request (GET requests/<requestID>) and this changes Request status to “authorization”. Now TachoCardAuthorizer starts querying Server about APDU packets directed from tachograph and GNSS/GPRS terminal.

At the same time Terminal starts querying GNSS/GPRS terminal about status and awaits the first APDU packet to be sent to company card.

Authorization

Authorization starts when GNSS/GPRS terminal reports that the first APDU packet is ready. Terminal passes it to Server, and when TachoCardAuthorizer asks about awaiting APDU packets (GET apdu/<requestID>/status) – Server confirms there is a APDU packet from device gpsID. Application reads the APDU packet (POST apdu/<gpsID>/dequeue), acknowledges its reception (POST apdu/<gpsID>/commit) and, after a while (fraction of second typically), sends back to Server a response from company card (POST apdu/<gpsID>).

Consecutive APDU packets delivered by GNSS/GPRS terminal should be numbered by Terminal (seqNumber) to keep packets order in case of GPRS connection problems. One packet can be retransmitted over GPRS (acknowledges are concern of the telemetric system), but it must not be delivered to GNSS/GPRS terminal twice. So if Terminal receives an APDU packet, which has already been sent to GNSS/GPRS terminal (recoginzd by seqNumber), it must be trashed. SeqNumber is passed with packet to Server and TachoCardAuthorizer. Reply of company card returns from TachoCardAuthorizer to Terminal with the same SeqNumber.

After Terminal passes the reply to GNSS/GPRS terminal, the Terminal continues querying about status and waits for next APDU packet from tachograph or for the end of authorization procedure.

Reply from company card to GNSS/GPRS terminal must be delivered within maximum 1 minute (typically no more than 10 seconds), otherwise GNSS/GPRS terminal would restart authorization and report new APDU packet to Terminal without waiting for previous reply.

After the third failed attempt, the GNSS/GPRS terminal reports authorization error and cancels the Request.

If some error is detected on the Server side (i.e. timeout, connection error, canceled by user), the Request status should be changed to “error”, which causes the TachoCardAuthorizer to cancel the authorization and get ready for further Requests.

The end of authorization

When tachograph accepts authorization of company card, the GNSS/GPRS terminal changes its status and the terminal has to forward this information to the Server so that the status of Request changes to “transferring”, indicating that the files are being downloaded from the tachograph. The TachoCardAuthorizer will be the last, knowing that the authorization ends as it continues to query the Server about Request status (GET apdu/<requestID>/status).

When the status changes, the TachoCardAuthorizer shows a message to the end user and start to query the Server for the next enqueued Request (GET queue). Anyway, TachoCardAuthorizer may now be closed and card reader detached.

Download files from tachograph

Since that moment, only the Terminal and the Server are engaged in Request.

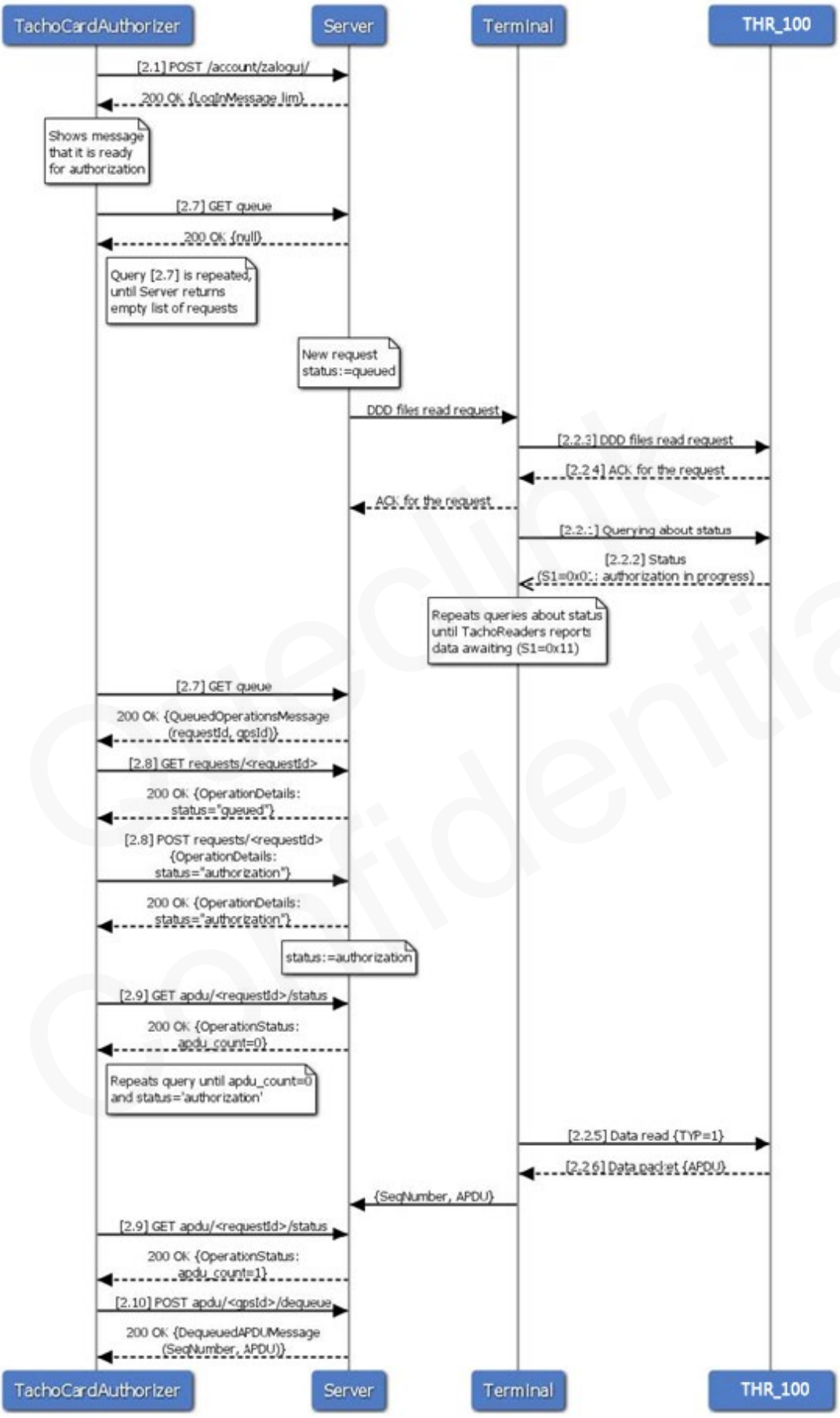
The Terminal continues to query Terminal about its status until it reports that files are ready. Then the Terminal must download files from the Terminal, including information about the name, length and checksum of the files, and forward them to the Server.

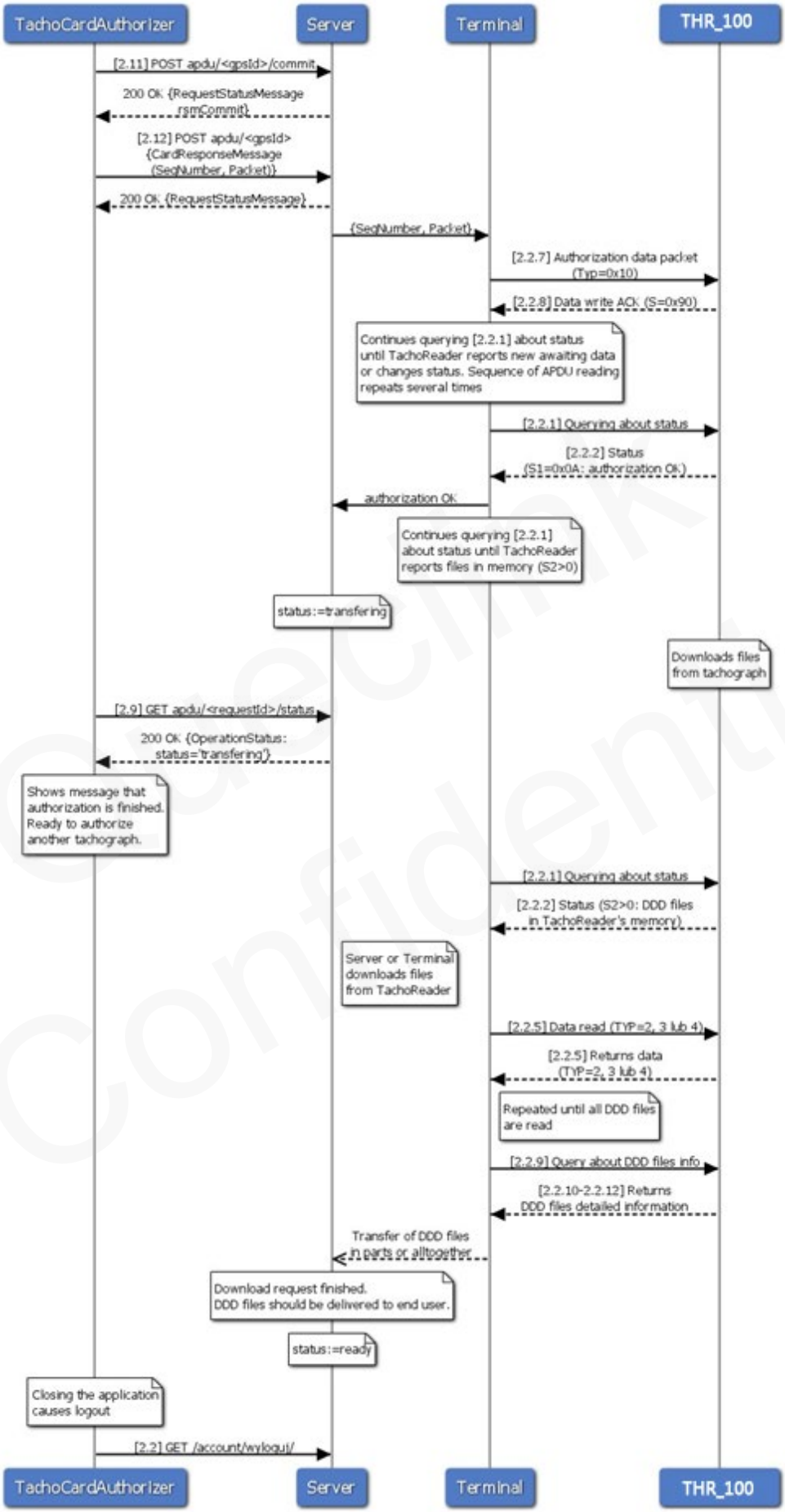
The Server stores the files or sends them to the end user, and the Request is over.

1.4.2. Sequence Diagram

Please check file “Sequence Diagram.pdf”.

Queclink Confidential


TachoCardAuthorizer
Server
Terminal
THR_100
[2.1] POST /account/zaloguj/
200 OK {LoginMessage lim}
Shows message that it is ready for authorization
[2.7] GET queue
200 OK {null}
Query [2.7] is repeated, until Server returns empty list of requests
New request status:=queued
DDD files read request
[2.2.3] DDD files read request
[2.2.4] ACK for the request
ACK for the request
[2.2.1] Querying about status
[2.2.2] Status (S1=0x01: authorization in progress)
Repeats queries about status until TachoReaders reports data awaiting (S1=0x11)
[2.7] GET queue
200 OK {QueuedOperationsMessage (requestId, gpsId)}
[2.8] GET requests/<requestId>
200 OK {OperationDetails: status="queued"}
[2.8] POST requests/<requestId> {OperationDetails: status="authorization"}
200 OK {OperationDetails: status="authorization"}
status:=authorization
[2.9] GET apdu/<requestId>/status
200 OK {OperationStatus: apdu_count=0}
Repeats query until apdu_count=0 and status='authorization'
[2.2.5] Data read {TYP=1}
[2.2.6] Data packet {APDU}
{SeqNumber, APDU}
[2.9] GET apdu/<requestId>/status
200 OK {OperationStatus: apdu_count=1}
[2.10] POST apdu/<gpsId>/dequeue
200 OK {DequeuedAPDUMessage (SeqNumber, APDU)}
TachoCardAuthorizer
Server
Terminal
THR_100



Queclink Confidential

Queclink
Rita Pan
2023.03.08