Commission Implementing Regulation (EU) 2016/799 of 18 March 2016 implementing Regulation (EU) No 165/2014 of the European Parliament and of the Council laying down the requirements for the construction, testing, installation, operation and repair of tachographs and their components (Text with EEA relevance)

ANNEX I C

Requirements for construction, testing, installation, and inspection

Appendix 2
TACHOGRAPH CARDS SPECIFICATION

1. INTRODUCTION

1.1. **Abbreviations**

For the purpose of this appendix, the following abbreviations apply.

AC	Access conditions
AES	Advanced Encryption Standard
AID	Application Identifier
ALW	Always
APDU	Application Protocol Data Unit (structure of a command)
ATR	Answer To Reset
AUT	Authenticated.
C6, C7	Contacts No 6 and 7 of the card as described in ISO/IEC 7816-2
cc	clock cycles
[F1CHA	Certificate Holder Authorisation**]**
CHV	Card holder Verification Information
CLA	Class byte of an APDU command
[F1DO	Data Object**]**
DSRC	Dedicated Short Range Communication
DF	Dedicated File. A DF can contain other files (EF or DF)
ECC	Elliptic Curve Cryptography
EF	Elementary File
etu	elementary time unit
G1	Generation 1
G2	Generation 2
IC	Integrated Circuit
ICC	Integrated Circuit Card
ID	Identifier
IFD	Interface Device
IFS	Information Field Size
IFSC	Information Field Size for the card
IFSD	Information Field Size Device (for the Terminal)
INS	Instruction byte of an APDU command
Lc	Length of the input data for a APDU command
Le	Length of the expected data (output data for a command)
MF	Master File (root DF)
NAD	Node Address used in T=1 protocol
NEV	Never
P1-P2	Parameter bytes
PIN	Personal Identification Number
PRO SM	Protected with secure messaging
PTS	Protocol Transmission Selection
RFU	Reserved for Future Use
RST	Reset (of the card)
SFID	Short EF Identifier
SM	Secure Messaging
SW1-SW2	Status bytes
TS	Initial ATR character
VPP	Programming Voltage
VU	Vehicle Unit

XXh	Value XX in hexadecimal notation
‘XXh’	Value XX in hexadecimal notation
||	Concatenation symbol 03||04=0304

Textual Amendments

F1 Inserted by Commission Implementing Regulation (EU) 2018/502 of 28 February 2018 amending Implementing Regulation (EU) 2016/799 laying down the requirements for the construction, testing, installation, operation and repair of tachographs and their components (Text with EEA relevance).

1.2. References

The following references are used in this Appendix:

ISO/IEC 7816-2	Identification cards — Integrated circuit cards — Part 2: Dimensions and location of the contacts. ISO/IEC 7816-2:2007.
ISO/IEC 7816-3	Identification cards — Integrated circuit cards — Part 3: Electrical interface and transmission protocols. ISO/IEC 7816-3:2006.
ISO/IEC 7816-4	Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange. ISO/IEC 7816-4:2013 + Cor 1: 2014.
ISO/IEC 7816-6	Identification cards — Integrated circuit cards — Part 6: Interindustry data elements for interchange. ISO/IEC 7816-6:2004 + Cor 1: 2006.
ISO/IEC 7816-8	Identification cards — Integrated circuit cards — Part 8: Commands for security operations. ISO/IEC 7816-8:2004.
ISO/IEC 9797-2	Information technology — Security techniques — Message Authentication Codes (MACs) — Part 2: Mechanisms using a dedicated hash-function. ISO/IEC 9797-2:2011

2. ELECTRICAL AND PHYSICAL CHARACTERISTICS

TCS_01 All electronic signals shall be in accordance with ISO/IEC 7816-3 unless specified otherwise.

TCS_02 The location and dimensions of the card contacts shall comply with the ISO/IEC 7816-2.

2.1. Supply Voltage and Current Consumption

TCS_03 The card shall work according to specifications within the consumption limits specified in ISO/IEC 7816-3.

TCS_04 The card shall work with Vcc = 3V (± 0,3V) or with Vcc = 5V (± 0,5V).

Voltage selection shall be performed according to ISO/IEC 7816-3.

2.2. Programming Voltage V_{pp}

TCS_05 The card shall not require a programming voltage at pin C6. It is expected that pin C6 is not connected in an IFD. Contact C6 may be connected to V_{cc} in the card but shall not be connected to ground. This voltage should not be interpreted in any case.

2.3. Clock generation and Frequency

TCS_06 The card shall operate within a frequency range of 1 to 5 MHz and may support higher frequencies. Within one card session the clock frequency may vary ± 2 %. The clock frequency is generated by the Vehicle Unit and not the card itself. The duty cycle may vary between 40 and 60 %.

TCS_07 Under conditions contained into the card file EF ICC, the external clock can be stopped. The first byte of the EF ICC file body codes the Clockstop mode conditions:

Low	**High**		
Bit 3	**Bit 2**	**Bit 1**	
0	0	1	Clockstop allowed, no preferred level
0	1	1	Clockstop allowed, high level preferred
1	0	1	Clockstop allowed, low level preferred
0	0	0	Clockstop not allowed
0	1	0	Clockstop only allowed on high level
1	0	0	Clockstop only allowed on low level

Bits 4 to 8 are not used.

2.4. I/O Contact

TCS_08 The I/O contact C7 is used to receive data from and to transmit data to the IFD. During operation only either the card or the IFD shall be in transmit mode. Should both units be in transmit mode no damage shall occur to the card. Unless transmitting, the card shall enter the reception mode.

2.5. States of the Card

TCS_09 The card works in two states while the supply voltage is applied:

Operation state while executing commands or interfacing with Digital Unit,

Idle state at all other times; in this state all data shall be retained by the card.

3. HARDWARE AND COMMUNICATION

3.1. Introduction

This paragraph describes the minimum functionality required by Tachograph cards and VUs to ensure correct operation and interoperability.

Tachograph cards are as compliant as possible with the available ISO/IEC applicable norms (especially ISO/IEC 7816). However, commands and protocols are fully described in order to specify some restricted usage or some differences if they exist. The commands specified are fully compliant with the referred norms except where indicated.

3.2. Transmission Protocol

TCS_10 The Transmission protocol shall be compliant with ISO/IEC 7816-3 for T = 0 and T = 1. In particular, the VU shall recognise waiting time extensions sent by the card.

3.2.1 *Protocols*

TCS_11 The card shall provide both protocol **T=0** and protocol **T=1**. In addition the card may support further contact-oriented protocols.

TCS_12 **T=0** is the default protocol, a **PTS** command is therefore necessary to change the protocol to **T=1**.

TCS_13 Devices shall support **direct convention** in both protocols: the direct convention is hence mandatory for the card.

TCS_14 The **Information Field Size Card** byte shall be presented at the ATR in character TA3. This value shall be at least ‘F0h’ (=240 bytes).

The following restrictions apply to the protocols:

TCS_15 **T=0**

— The interface device shall support an answer on I/O after the rising edge of the signal on RST from 400 cc.

— The interface device shall be able to read characters separated with 12 etu.

— The interface device shall read an erroneous character and its repetition if separated with 13 etu. If an erroneous character is detected, the Error signal on I/O can occur between 1 etu and 2 etu. The device shall support a 1 etu delay.

— The interface device shall accept a 33 bytes ATR (TS+32)

— If TC1 is present in the ATR, the Extra Guard Time shall be present for characters sent by the interface device although characters sent by the card can still be separated with 12 etu. This is also true for the ACK character sent by the card after a P3 character emitted by the interface device.

— The interface device shall take into account a NUL character emitted by the card.

— The interface device shall accept the complementary mode for ACK.

— The get-response command cannot be used in chaining mode to get a data which length could exceed 255 bytes.

TCS_16 **T=1**

— NAD byte: not used (NAD shall be set to ‘00’).

— S-block ABORT: not used.

— S-block VPP state error: not used.

— The total chaining length for a data field will not exceed 255 bytes (to be ensured by the IFD).

— The Information Field Size Device (IFSD) shall be indicated by the IFD immediately after the ATR: the IFD shall transmit the S-Block IFS request after the ATR and the card shall send back S-Block IFS. The recommended value for IFSD is 254 bytes.

— The card will not ask for an IFS readjustment.

3.2.2 *ATR*

TCS_17 The device checks ATR bytes, according to ISO/IEC 7816-3. No verification shall be done on ATR Historical Characters.

Example of Basic Biprotocol ATR according to ISO/IEC 7816-3

Character	Value	Remarks

TS	‘3Bh’	Indicates direct convention.
T0	‘85h’	TD1 present; 5 historical bytes are presents.
TD1	‘80h’	TD2 present; T=0 to be used
TD2	‘11h’	TA3 present; T=1 to be used
TA3	‘XXh’ (at least ‘F0h’)	Information Field Size Card (IFSC)
TH1 to TH5	‘XXh’	Historical characters
TCK	‘XXh’	Check Character (exclusive OR)

TCS_18 After the Answer To Reset (ATR), the Master File (MF) is implicitly selected and becomes the Current Directory.

3.2.3 *PTS*

TCS_19 The default Protocol is T=0. To set the T=1 protocol, a PTS (also known as PPS) must be sent to the card by the device.

TCS_20 As both T=0 and T=1 protocols are mandatory for the card, the basic PTS for protocol switching is mandatory for the card.

The PTS can be used, as indicated in ISO/IEC 7816-3, to switch to higher baud rates than the default one proposed by the card in the ATR if any (TA(1) byte).

Higher baud rates are optional for the card.

TCS_21 If no other baud rate than the default one are supported (or if the selected baud rate is not supported), the card shall respond to the PTS correctly according to ISO/IEC 7816-3 by omitting the PPS1 byte.

Examples of basic PTS for protocol selection are the following:

Character	**Value**	**Remarks**
PPSS	‘FFh’	The Initiate Character.
PPS0	‘00h’ or ‘01h’	PPS1 to PPS3 are not present; ‘00h’ to select T0, ‘01h’ to select T1.
PK	‘XXh’	Check: ‘XXh’ = ‘FFh’ if Character PPS0 = ‘00h’, ‘XXh’ = ‘FEh’ if PPS0 = ‘01h’.

3.3. **Access Rules**

TCS_22 An access rule specifies for an access mode, i.e. command, the corresponding security conditions. If these security conditions are fulfilled the corresponding command is processed.

TCS_23 The following security conditions are used for the tachograph card:

Abbreviation	Meaning
ALW	The action is always possible and can be executed without any restriction. Command and response APDU are sent in plain text, i.e. without secure messaging.
NEV	The action is never possible.
PLAIN-C	The command APDU is sent in plain, i.e. without secure messaging.
PWD	The action may only be executed if the workshop card PIN has been successfully verified, i.e. if the card internal security status 'PIN_Verified' is set. The command must be sent without secure messaging.
EXT-AUT-G1	The action may only be executed if the External Authenticate command for the generation 1 authentication (see also Appendix 11 Part A) has been successfully performed.
SM-MAC-G1	The APDU (command and response) must be applied with generation 1 secure messaging in authentication-only mode (see Appendix 11 Part A).
SM-C-MAC-G1	The command APDU must be applied with generation 1 secure messaging in authentication only mode (see Appendix 11 Part A).
SM-R-ENC-G1	The response APDU must be applied with generation 1 secure messaging in encryption mode (see Appendix 11 Part A), i.e. no message authentication code is returned.
SM-R-ENC-MAC-G1	The response APDU must be applied with generation 1 secure messaging in encrypt-then-authenticate mode (see Appendix 11 Part A).
SM-MAC-G2	The APDU (command and response) must be applied with generation 2 secure messaging in authentication-only mode (see Appendix 11 Part B).
SM-C-MAC-G2	The command APDU must be applied with generation 2 secure messaging in authentication only mode (see Appendix 11 Part B).
SM-R-ENC-MAC-G2	The response APDU must be applied with generation 2 secure messaging in encrypt-

	then-authenticate mode (see Appendix 11 Part B).

[F2TCS_24These security conditions can be linked in the following ways:

AND	:	All security conditions must be fulfilled
OR	:	At least one security condition must be fulfilled

The access rules for the file system, i.e. the SELECT, READ BINARY and UPDATE BINARY command, are specified in chapter 4. The access rules for the remaining commands are specified in the following tables. The term 'not applicable' is used if there is no requirement to support the command. In this case the command may or may not be supported, but the access condition is out of scope.]

Textual Amendments

F2 Substituted by Commission Implementing Regulation (EU) 2018/502 of 28 February 2018 amending Implementing Regulation (EU) 2016/799 laying down the requirements for the construction, testing, installation, operation and repair of tachographs and their components (Text with EEA relevance).

TCS_25 In the DF Tachograph G1 application the following access rules are used:

[F2**Command**	**Driver Card**	**Workshop Card**	**Control Card**	**Company Card**
External Authenticate				
— For generation 1 authentication	ALW	ALW	ALW	ALW
— For generation 2 authentication	ALW	PWD	ALW	ALW
Internal Authenticate	ALW	PWD	ALW	ALW
General Authenticate	ALW	ALW	ALW	ALW
Get Challenge	ALW	ALW	ALW	ALW
MSE:SET AT	ALW	ALW	ALW	ALW
MSE:SET DST	ALW	ALW	ALW	ALW
Process DSRC Message	Not applicable	Not applicable	Not applicable	Not applicable
PSO: Compute Digital Signature	ALW OR SM-MAC-G2	ALW OR SM-MAC-G2	Not applicable	Not applicable

PSO: Hash	Not applicable	Not applicable	ALW	Not applicable
PERFORM HASH of FILE	ALW OR SM-MAC-G2	ALW OR SM-MAC-G2	Not applicable	Not applicable
PSO: Verify Certificate	ALW	ALW	ALW	ALW
PSO: Verify Digital Signature	Not applicable	Not applicable	ALW	Not applicable
Verify	Not applicable	ALW	Not applicable	Not applicable**]**

TCS_26 In the DF Tachograph_G2 application the following access rules are used:

[F2Command	**Driver Card**	**Workshop Card**	**Control Card**	**Company Card**
External Authenticate				
— For generation 1 authentication	Not applicable	Not applicable	Not applicable	Not applicable
— For generation 2 authentication	ALW	PWD	ALW	ALW
Internal Authenticate	Not applicable	Not applicable	Not applicable	Not applicable
General Authenticate	ALW	ALW	ALW	ALW
Get Challenge	ALW	ALW	ALW	ALW
MSE:SET AT	ALW	ALW	ALW	ALW
MSE:SET DST	ALW	ALW	ALW	ALW
Process DSRC Message	Not applicable	ALW	ALW	Not applicable
PSO: Compute Digital Signature	ALW OR SM-MAC-G2	ALW OR SM-MAC-G2	Not applicable	Not applicable
PSO: Hash	Not applicable	Not applicable	ALW	Not applicable
PERFORM HASH of FILE	ALW OR SM-MAC-G2	ALW OR SM-MAC-G2	Not applicable	Not applicable
PSO: Verify Certificate	ALW	ALW	ALW	ALW

PSO: Verify Digital Signature	Not applicable	Not applicable	ALW	Not applicable
Verify	Not applicable	ALW	Not applicable	Not applicable]

TCS_27 In the MF the following access rules are used:

[F2**Command**	**Driver Card**	**Workshop Card**	**Control Card**	**Company Card**
External Authenticate				
— For generation 1 authentication	Not applicable	Not applicable	Not applicable	Not applicable
— For generation 2 authentication	ALW	PWD	ALW	ALW
Internal Authenticate	Not applicable	Not applicable	Not applicable	Not applicable
General Authenticate	ALW	ALW	ALW	ALW
Get Challenge	ALW	ALW	ALW	ALW
MSE:SET AT	ALW	ALW	ALW	ALW
MSE:SET DST	ALW	ALW	ALW	ALW
Process DSRC Message	Not applicable	Not applicable	Not applicable	Not applicable
PSO: Compute Digital Signature	Not applicable	Not applicable	Not applicable	Not applicable
PSO: Hash	Not applicable	Not applicable	Not applicable	Not applicable
PERFORM HASH of FILE	Not applicable	Not applicable	Not applicable	Not applicable
PSO: Verify Certificate	ALW	ALW	ALW	ALW
PSO: Verify Digital Signature	Not applicable	Not applicable	Not applicable	Not applicable
Verify	Not applicable	ALW	Not applicable	Not applicable]

TCS_28 A tachograph card may or may not accept a command with a higher level of security than the one specified in the security conditions. I.e. if the security condition is ALW (or PLAIN-C) the card may accept a command with secure messaging (encryption and / or authentication mode). If the security condition requires secure messaging with authentication mode, the tachograph card may accept a command with secure messaging of the same generation in authentication and encryption mode.

Note: The command descriptions provide more information on the support of the commands for the different tachograph card types and the different DFs.

3.4. Commands and error codes overview

Commands and file organisation are deduced from and complies with ISO/IEC 7816-4.

This section describes the following APDU command-response pairs. The command variants which are supported by a generation 1 and 2 application are specified in the corresponding command descriptions.

Command	**INS**
SELECT	‘A4h’
READ BINARY	‘B0h’, ‘B1h’
UPDATE BINARY	‘D6h’, ‘D7h’
GET CHALLENGE	‘84h’
VERIFY	‘20h’
GET RESPONSE	‘C0h’
PERFORM SECURITY OPERATION	‘2Ah’
— VERIFY CERTIFICATE	
— COMPUTE DIGITAL SIGNATURE	
— VERIFY DIGITAL SIGNATURE	
— HASH	
— PERFORM HASH OF FILE	
— PROCESS DSRC MESSAGE	
INTERNAL AUTHENTICATE	‘88h’
EXTERNAL AUTHENTICATE	‘82h’
MANAGE SECURITY ENVIRONMENT	‘22h’
— SET DIGITAL SIGNATURE TEMPLATE	
— SET AUTHENTICATION TEMPLATE	
GENERAL AUTHENTICATE	‘86h’

[F2TCS_29 The status words SW1 SW2 are returned in any response message and denote the processing state of the command.

SW1	SW2	Meaning
90	00	Normal processing.
61	XX	Normal processing. XX = number of response bytes available.
62	81	Warning processing. Part of returned data may be corrupted
63	00	Authentication failed (Warning)
63	CX	Wrong CHV (PIN). Remaining attempts counter provided by 'X'.
64	00	Execution error - State of non-volatile memory unchanged. Integrity error.
65	00	Execution error - State of non-volatile memory changed
65	81	Execution error - State of non-volatile memory changed – Memory failure
66	88	Security error wrong cryptographic checksum (during Secure Messaging) or wrong certificate (during certificate verification) or wrong cryptogram (during external authentication) or wrong signature (during signature verification)
67	00	Wrong length (wrong Lc or Le)
68	83	Last command of the chain expected
69	00	Forbidden command (no response available in T=0)
69	82	Security status not satisfied.

69	83	Authentication method blocked.
69	85	Conditions of use not satisfied.
69	86	Command not allowed (no current EF).
69	87	Expected Secure Messaging Data Objects missing
69	88	Incorrect Secure Messaging Data Objects
6A	80	Incorrect parameters in data field
6A	82	File not found.
6A	86	Wrong parameters P1-P2.
6A	88	Referenced data not found.
6B	00	Wrong parameters (offset outside the EF).
6C	XX	Wrong length, SW2 indicates the exact length. No data field is returned.
6D	00	Instruction code not supported or invalid.
6E	00	Class not supported.
6F	00	— Other checking errors

Additional status words as defined in ISO/IEC 7816-4 can be returned, if their behaviour is not explicitly mentioned in this appendix.

For example the following status words can be optionally returned:

6881: Logical channel not supported

6882: Secure messaging not supported**]**

TCS_30 If more than one error condition is fulfilled in one command APDU the card may return any of the appropriate status words.

3.5. Command descriptions

The mandatory commands for the Tachograph cards are described in this chapter.

Additional relevant details, related to cryptographic operations involved, are given in Appendix 11 Common security mechanisms for Tachograph Generation 1 and Generation 2.

All commands are described independently of the used protocol (T=0 or T=1). The APDU bytes CLA, INS, P1, P2, Lc and Le are always indicated. If Lc or Le is not needed for the described command, the associated length, value and description are empty.

TCS_31 If both length bytes (Lc and Le) are requested, the described command has to be split in two parts if the IFD is using protocol T=0: the IFD sends the command as described with P3=Lc + data and then sends a GET RESPONSE (see § 3.5.6) command with P3=Le.

TCS_32 If both length bytes are requested, and Le=0 (secure messaging):

— When using protocol T=1, the card shall answer to Le=0 by sending all available output data.

— When using protocol T=0, the IFD shall send the first command with P3=Lc + data, the card shall answer (to this implicit Le=0) by the Status bytes '**61La**', where La is the number of response bytes available. The IFD shall then generate a GET RESPONSE command with P3 = La to read the data.

TCS_33 A tachograph card may support extended length fields according to ISO/IEC 7816-4 as an optional feature. A tachograph card that supports extended length fields shall

— Indicate the extended length field support in the ATR

— Provide the supported buffer sizes by means of the extended length information in the EF ATR/INFO see TCS_146.

— Indicate whether it supports extended length fields for T = 1 and / or T = 0 in the EF Extended Length, see TCS_147.

— Support extended length fields for the tachograph application generation 1 and 2.

Notes:

All commands are specified for short length fields. The usage of extended length APDUs is clear from ISO/IEC 7816-4.

In general the commands are specified for the plain mode, i.e. without secure messaging, as the secure messaging layer is specified in Appendix 11. It is clear from the access rules for a command whether the command shall support secure messaging or not and whether the command shall support generation 1 and / or generation 2 secure messaging. Some command variants are described with secure messaging to illustrate the usage of secure messaging.

TCS_34 The VU shall perform the complete generation 2 VU — card mutual authentication protocol for a session including the certificate verification (if required) either in the DF Tachograph, the DF Tachograph_G2 or the MF.

3.5.1 *SELECT*

This command is compliant with ISO/IEC 7816-4, but has a restricted usage compared to the command defined in the norm.

The SELECT command is used:

— to select an application DF (selection by name must be used)

— to select an elementary file corresponding to the submitted file ID

3.5.1.1 *Selection by name (AID)*

This command allows selecting an application DF in the card.

TCS_35 This command can be performed from anywhere in the file structure (after the ATR or at any time).

TCS_36 The selection of an application resets the current security environment. After performing the application selection, no current public key is selected anymore. The EXT-AUT-G1 access condition is also lost. If the command was performed without secure messaging, the former secure messaging session keys are no longer available.

TCS_37 **Command Message**

Byte	Length	Value	Description
CLA	1	‘00h’	
INS	1	‘A4h’	
P1	1	‘04h’	Selection by name (AID)
P2	1	‘0Ch’	No response expected
Lc	1	‘NNh’	Number of bytes sent to the card (length of the AID): ‘06h’ for the Tachograph application
#6-#(5+NN)	NN	‘XX..XXh’	AID: ‘FF 54 41 43 48 4F’ for the Generation 1 tachograph application AID: ‘FF 53 4D 52 44 54’ for the Generation 2 tachograph application

No response to the SELECT command is needed (Le absent in T=1, or no response asked in T=0).

TCS_38 **Response Message (no response asked)**

Byte	Length	Value	Description
SW	2	‘XXXXh’	Status Words (SW1,SW2)

— If the command is successful, the card returns ‘**9000**’.

— If the application corresponding with the AID is not found, the processing state returned is ‘**6A82**’.

— In T=1, if the byte Le is present, the state returned is ‘**6700**’.

— In T=0, if a response is asked after the SELECT command, the state returned is ‘**6900**’.

— **[**[F2]If the selected application is considered to be corrupted (integrity error is detected within the file attributes), the processing state returned is ‘**6400**’ or ‘**6500**’.**]**

3.5.1.2 *Selection of an Elementary File using its File Identifier*

TCS_39 **Command Message**

TCS_40 A tachograph card shall support the generation 2 secure messaging as specified in Appendix 11 Part B for this command variant.

Byte	**Length**	**Value**	**Description**
CLA	1	‘00h’	
INS	1	‘A4h’	
P1	1	‘02h’	Selection of an EF under the current DF
P2	1	‘0Ch’	No response expected
Lc	1	‘02h’	Number of bytes sent to the card
#6-#7	2	‘XXXXh’	File Identifier

No response to the SELECT command is needed (Le absent in T=1, or no response asked in T=0).

TCS_41 **Response Message (no response asked)**

Byte	**Length**	**Value**	**Description**
SW	2	‘XXXXh’	Status Words (SW1,SW2)

— If the command is successful, the card returns ‘**9000**’.

— If the file corresponding with the file identifier is not found, the processing state returned is ‘**6A82**’.

— In T=1, if the byte Le is present, the state returned is ‘**6700**’.

— In T=0, if a response is asked after the SELECT command, the state returned is ‘**6900**’.

— **[**[F2]If the selected file is considered to be corrupted (integrity error is detected within the file attributes), the processing state returned is ‘**6400**’ or ‘**6500**’.**]**

3.5.2 *READ BINARY*

This command is compliant with ISO/IEC 7816-4, but has a restricted usage compared to the command defined in the norm.

The READ BINARY command is used to read data from a transparent file.

The response of the card consists of returning the data read, optionally encapsulated in a secure messaging structure.

3.5.2.1 *Command with offset in P1-P2*

This command enables the IFD to read data from the EF currently selected, without secure messaging.

Note: This command without secure messaging can only be used to read a file that supports the ALW security condition for the Read access mode.

TCS_42 **Command Message**

Byte	Length	Value	Description
CLA	1	‘00h’	
INS	1	‘B0h’	Read Binary
P1	1	‘XXh’	Offset in bytes from the beginning of the file: Most Significant Byte
P2	1	‘XXh’	Offset in bytes from the beginning of the file: Least Significant Byte
Le	1	‘XXh’	Length of data expected. Number of Bytes to be read.

Note: bit 8 of P1 must be set to 0.

TCS_43 **Response Message**

Byte	Length	Value	Description
#1-#X	X	‘XX..XXh’	Data read
SW	2	‘XXXXh’	Status Words (SW1,SW2)

- If the command is successful, the card returns ‘**9000**’.
- If no EF is selected, the processing state returned is ‘**6986**’.
- If the security conditions of the selected file are not satisfied, the command is interrupted with ‘**6982**’.
- If the Offset is not compatible with the size of the EF (Offset > EF size), the processing state returned is ‘**6B00**’.
- If the size of the data to be read is not compatible with the size of the EF (Offset + Le > EF size) the processing state returned is ‘**6700**’ or ‘**6Cxx**’ where ‘xx’ indicates the exact length.
- **[**[F2]If an integrity error is detected within the file attributes, the card shall consider the file as corrupted and unrecoverable, the processing state returned is ‘**6400**’ or ‘**6500**’.**]**
- If an integrity error is detected within the stored data, the card shall return the demanded data, and the processing state returned is ‘**6281**’.

3.5.2.1.1 *Command with secure messaging (examples)*

This command enables the IFD to read data from the EF currently selected with secure messaging, in order to verify the integrity of the data received and to protect the confidentiality of the data if the security condition SM-R-ENC-MAC-G1 (generation 1) or SM-R-ENC-MAC-G2 (generation 2) is applied.

TCS_44 **Command Message**

Byte	Length	Value	Description

CLA	1	‘0Ch’	Secure Messaging asked
INS	1	‘B0h’	Read Binary
P1	1	‘XXh’	P1 (offset in bytes from the beginning of the file): Most Significant Byte
P2	1	‘XXh’	P2 (offset in bytes from the beginning of the file): Least Significant Byte
Lc	1	‘XXh’	Length of input data for secure messaging
#6	1	‘97h’	T_{LE}: Tag for expected length specification.
#7	1	‘01h’	L_{LE}: Length of expected length
#8	1	‘NNh’	Expected length specification (original Le): Number of Bytes to be read
#9	1	‘8Eh’	T_{CC}: Tag for cryptographic checksum
#10	1	‘XXh’	L_{CC}: Length of following cryptographic checksum ‘04h’ for Generation 1 secure messaging (see Appendix 11 Part A) ‘08h’, ‘0Ch’ or ‘10h’ depending on AES key length for Generation 2 secure messaging (see Appendix 11 Part B)

#11-#(10+L)	L	'XX..XXh'	Cryptographic checksum
Le	1	'00h'	As specified in ISO/IEC 7816-4

TCS_45 **Response Message if SM-R-ENC-MAC-G1 (generation 1) / SM-R-ENC-MAC-G2 (generation 2) is not required and if Secure Messaging input format is correct:**

[F2**Byte**	**Length**	**Value**	**Description**
#1	1	'81h'	T_{PV}: Tag for plain value data
#2	L	'NNh' or '81 NNh'	L_{PV}: length of returned data (=original Le). L is 2 bytes if L_{PV}>127 bytes.
#(2+L) - #(1+L+NN)	NN	'XX..XXh'	Plain Data value
#(2+L+NN)	1	'99h'	Tag for Processing Status (SW1-SW2) – optional for generation 1 secure messaging
#(3+L+NN)	1	'02h'	Length of Processing Status – optional for generation 1 secure messaging
#(4+L+NN) - #(5+L+NN)	2	'XX XXh'	Processing Status of the unprotected response APDU – optional for generation 1 secure messaging
#(6+L+NN)	1	'8Eh'	TCC: Tag for cryptographic checksum
#(7+L+NN)	1	'XXh'	LCC: Length of following cryptographic checksum '04h' for Generation 1 secure messaging (see Appendix 11 Part A)

			'08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix 11 Part B)
#(8+L+NN)-#(7+M+L+NN)	M	'XX..XXh'	Cryptographic checksum
SW	2	'XXXXh'	Status Words (SW1,SW2)**]**

TCS_46 **Response Message if SM-R-ENC-MAC-G1 (generation 1) / SM-R-ENC-MAC-G2 (generation 2) is required and if Secure Messaging input format is correct:**

[F2**Byte**	**Length**	**Value**	**Description**
#1	1	'87h'	$T_{PI\ CG}$: Tag for encrypted data (cryptogram)
#2	L	'MMh' or '81 MMh'	$L_{PI\ CG}$: length of returned encrypted data (different of original Le of the command due to padding). L is 2 bytes if LPI CG > 127 bytes.
#(2+L)-#(1+L+MM)	MM	'01XX..XXh'	Encrypted Data: Padding Indicator and cryptogram
#(2+L+MM)	1	'99h'	Tag for Processing Status (SW1-SW2) – optional for generation 1 secure messaging
#(3+L+MM)	1	'02h'	Length of Processing Status – optional for generation 1 secure messaging
#(4+L+MM) - #(5+L+MM)	2	'XX XXh'	Processing Status of the unprotected response APDU – optional for

			generation 1 secure messaging
#(6+L+MM)	1	‘8Eh’	TCC: Tag for cryptographic checksum
#(7+L+MM)	1	‘XXh’	LCC: Length of following cryptographic checksum ‘04h’ for Generation 1 secure messaging (see Appendix 11 Part A) ‘08h’, ‘0Ch’ or ‘10h’ depending on AES key length for Generation 2 secure messaging (see Appendix 11 Part B)
#(8+L+MM)-#(7+N+L+MM)	N	‘XX..XXh’	Cryptographic checksum
SW	2	‘XXXXh’	Status Words (SW1,SW2)**]**

The READ BINARY command may return regular processing states listed in TCS_43 under Tag ‘99h’ as described in TCS_59 using the secure messaging response structure.

Additionally, some errors specifically related to secure messaging can happen. In that case, the processing state is simply returned, with no secure messaging structure involved:

TCS_47 **Response Message if incorrect Secure Messaging input format**

Byte	Length	Value	Description
SW	2	‘XXXXh’	Status Words (SW1,SW2)

— If no current session key is available, the processing state ‘**6A88**’ is returned. It happens either if the session key has not already been generated or if the session key validity has expired (in this case the IFD must re-run a mutual authentication process to set a new session key).

— If some expected data objects (as specified above) are missing in the secure messaging format, the processing state '**6987**' is returned: this error happens if an expected tag is missing or if the command body is not properly constructed.

— If some data objects are incorrect, the processing state returned is '**6988**': this error happens if all the required tags are present but some lengths are different from the ones expected.

— If the verification of the cryptographic checksum fails, the processing state returned is '**6688**'.

3.5.2.2 *Command with short EF (Elementary File) identifier*

This command variant enables the IFD to select an EF by means of a short EF identifier and read data from this EF.

TCS_48 A tachograph card shall support this command variant for all Elementary Files with a specified short EF identifier. These short EF identifiers are specified in chapter 4.

TCS_49 **Command Message**

Byte	**Length**	**Value**	**Description**
CLA	1	'00h'	
INS	1	'B0h'	Read Binary
P1	1	'XXh'	Bit 8 is set to 1 Bit 7 and 6 are set to 00 Bit 5 — 1 encode the short EF identifier of the corresponding EF
P2	1	'XXh'	Encodes an offset from 0 to 255 bytes in the EF referenced by P1
Le	1	'XXh'	Length of data expected. Number of Bytes to be read.

Note: The short EF identifiers used for the Generation 2 tachograph application are specified in chapter 4.

If P1 encodes a short EF identifier and the command is successful, the identified EF becomes the currently selected EF (current EF).

TCS_50 **Response Message**

Byte	**Length**	**Value**	**Description**
#1-#L	L	'XX..XXh'	Data read
SW	2	'XXXXh'	Status Words (SW1,SW2)

— If the command is successful, the card returns '**9000**'.

— If the file corresponding with the short EF identifier is not found, the processing state returned is ‘**6A82**’.

— If the security conditions of the selected file are not satisfied, the command is interrupted with ‘**6982**’.

— If the Offset is not compatible with the size of the EF (Offset > EF size), the processing state returned is ‘**6B00**’.

— If the size of the data to be read is not compatible with the size of the EF (Offset + Le > EF size) the processing state returned is ‘**6700**’ or ‘**6Cxx**’ where ‘xx’ indicates the exact length.

— **[**F2If an integrity error is detected within the file attributes, the card shall consider the file as corrupted and unrecoverable, the processing state returned is ‘**6400**’ or ‘**6500**’.**]**

— If an integrity error is detected within the stored data, the card shall return the demanded data, and the processing state returned is ‘**6281**’.

3.5.2.3 *Command with odd instruction byte*

This command variant enables the IFD to read data from an EF with 32 768 bytes or more.

TCS_51 A tachograph card which supports EFs with 32 768 bytes or more shall support this command variant for these EFs. A tachograph card may or may not support this command variant for other EFs with the exception of the EF Sensor_Installation_Data see TCS_156 and TCS_160.

TCS_52 **Command Message**

Byte	Length	Value	Description
CLA	1	‘00h’	
INS	1	‘B1h’	Read Binary
P1	1	‘00h’	Current EF
P2	1	‘00h’	
Lc	1	‘NNh’	Lc Length of offset data object.
#6-#(5+NN)	NN	‘XX..XXh’	Offset data object: Tag ‘54h’ Length ‘01h’ or ‘02h’ Value offset
[F2Le	1	'XXh'	As specified in ISO/IEC 7816-4**]**

The IFD shall encode the offset data object's length with a minimum possible number of octets, i.e. using the length byte ‘01h’ the IFD shall encode an offset from 0 to 255 and using the length byte ‘02h’ an offset from ‘256’ up to ‘65 535’ bytes.

[F1In case of T = 0 the card assumes the value Le = ‘00h’ if no secure messaging is applied.

In case of T = 1 the processing state returned is ‘6700’ if Le=‘01h’.**]**

TCS_53 **Response Message**

Byte	Length	Value	Description
#1-#L	L	'XX..XXh'	Data read encapsulated in a discretionary data object with tag '53h'.
SW	2	'XXXXh'	Status Words (SW1,SW2)

— If the command is successful, the card returns '**9000**'.

— If no EF is selected, the processing state returned is '**6986**'.

— If the security conditions of the selected file are not satisfied, the command is interrupted with '**6982**'.

— If the Offset is not compatible with the size of the EF (Offset > EF size), the processing state returned is '**6B00**'.

— If the size of the data to be read is not compatible with the size of the EF (Offset + Le > EF size) the processing state returned is '**6700**' or '**6Cxx**' where 'xx' indicates the exact length.

— **[**[F2]If an integrity error is detected within the file attributes, the card shall consider the file as corrupted and unrecoverable, the processing state returned is '**6400**' or '**6500**'.**]**

— If an integrity error is detected within the stored data, the card shall return the demanded data, and the processing state returned is '**6281**'.

3.5.2.3.1 *Command with secure messaging (example)*

The following example illustrates the usage of secure messaging if the security condition SM-MAC-G2 applies.

TCS_54 Command message

Byte	Length	Value	Description
CLA	1	'0Ch'	Secure Messaging asked
INS	1	'B1h'	Read Binary
P1	1	'00h'	Current EF
P2	1	'00h'	
Lc	1	'XXh'	Length of the secured data field
#6	1	'B3h'	Tag for plain value data encoded in BER-TLV
#7	1	'NNh'	L_{PV}: length of transmitted data
#(8)-#(7+NN)	NN	'XX..XXh'	Plain Data encoded in BER-TLV, i.e. the offset data object with tag '54'

#(8+NN)	1	'97h'	T_{LE}: Tag for expected length specification.
#(9+NN)	1	'01h'	L_{LE}: Length of expected length
#(10+NN)	1	'XXh'	Expected length specification (original Le): Number of bytes to be read
#(11+NN)	1	'8Eh'	T_{CC}: Tag for cryptographic checksum
#(12+NN)	1	'XXh'	L_{CC}: Length of following cryptographic checksum '08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix 11 Part B)
#(13+NN)-#(12+M +NN)	M	'XX..XXh'	Cryptographic checksum
Le	1	'00h'	As specified in ISO/ IEC 7816-4

TCS_55 Response message if the command is successful

Byte	**Length**	**Value**	**Description**
#1	1	'B3h'	Plain Data encoded in BER-TLV
#2	L	'NNh' or '81 NNh'	L_{PV}: length of returned data (=original Le). L is 2 bytes if L_{PV}>127 bytes.
#(2+L)-#(1+L+NN)	NN	'XX..XXh'	Plain Data value encoded in BER-TLV, i.e. data read encapsulated in a discretionary data object with tag '53h'.
#(2+L+NN)	1	'99h'	Processing Status of the unprotected response APDU

#(3+L+NN)	1	'02h'	Length of Processing Status
#(4+L+NN) — #(5+L+NN)	2	'XX XXh'	Processing Status of the unprotected response APDU
#(6+L+NN)	1	'8Eh'	T_{CC}: Tag for cryptographic checksum
#(7+L+NN)	1	'XXh'	L_{CC}: Length of following cryptographic checksum '08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix 11 Part B)
#(8+L+NN)-#(7+M+L+NN)	M	'XX..XXh'	Cryptographic checksum
SW	2	'XXXXh'	Status Words (SW1,SW2)

3.5.3 *UPDATE BINARY*

This command is compliant with ISO/IEC 7816-4, but has a restricted usage compared to the command defined in the norm.

The UPDATE BINARY command message initiates the update (erase + write) of the bits already present in an EF binary with the bits given in the command APDU.

3.5.3.1 *Command with offset in P1-P2*

This command enables the IFD to write data into the EF currently selected, without the card verifying the integrity of data received.

Note: This command without secure messaging can only be used to update a file that supports the ALW security condition for the Update access mode.

TCS_56 **Command Message**

Byte	**Length**	**Value**	**Description**
CLA	1	'00h'	
INS	1	'D6h'	Update Binary
P1	1	'XXh'	Offset in bytes from the beginning of the file: Most Significant Byte

P2	1	'XXh'	Offset in bytes from the beginning of the file: Least Significant Byte
Lc	1	'NNh'	Lc Length of data to Update. Number of bytes to be written.
#6-#(5+NN)	NN	'XX..XXh'	Data to be written

Note: bit 8 of P1 must be set to 0.

TCS_57 **Response Message**

Byte	Length	Value	Description
SW	2	'XXXXh'	Status Words (SW1,SW2)

— If the command is successful, the card returns '**9000**'.

— If no EF is selected, the processing state returned is '**6986**'.

— If the security conditions of the selected file are not satisfied, the command is interrupted with '**6982**'.

— If the Offset is not compatible with the size of the EF (Offset > EF size), the processing state returned is '**6B00**'.

— If the size of the data to be written is not compatible with the size of the EF (Offset + Lc > EF size) the processing state returned is '**6700**'.

— If an integrity error is detected within the file attributes, the card shall consider the file as corrupted and unrecoverable, the processing state returned is '**6400**' or '**6500**'.

— If writing is unsuccessful, the processing state returned is '**6581**'.

3.5.3.1.1 *Command with secure messaging (examples)*

This command enables the IFD to write data into the EF currently selected, with the card verifying the integrity of data received. As no confidentiality is required, the data are not encrypted.

TCS_58 **Command Message**

Byte	Length	Value	Description
CLA	1	'0Ch'	Secure Messaging asked
INS	1	'D6h'	Update Binary
P1	1	'XXh'	Offset in bytes from the beginning of the file: Most Significant Byte
P2	1	'XXh'	Offset in bytes from the beginning of the file:

			Least Significant Byte
Lc	1	‘XXh’	Length of the secured data field
#6	1	‘81h’	T_{PV}: Tag for plain value data
#7	L	‘NNh’ or ‘81 NNh’	L_{PV}: length of transmitted data. L is 2 bytes if L_{PV} > 127 bytes.
#(7+L)-#(6+L+NN)	NN	‘XX..XXh’	Plain Data value (Data to be written)
#(7+L+NN)	1	‘8Eh’	T_{CC}: Tag for cryptographic checksum
#(8+L+NN)	1	‘XXh’	L_{CC}: Length of following cryptographic checksum‘04h’ for Generation 1 secure messaging (see Appendix 11 Part A) ‘08h’, ‘0Ch’ or ‘10h’ depending on AES key length for Generation 2 secure messaging (see Appendix 11 Part B)
#(9+L+NN)-#(8+M +L+NN)	M	‘XX..XXh’	Cryptographic checksum
Le	1	‘00h’	As specified in ISO/ IEC 7816-4

TCS_59 **Response message if correct Secure Messaging input format**

Byte	Length	Value	Description
#1	1	‘99h’	T_{SW}: Tag for Status Words (to be protected by CC)
#2	1	‘02h’	L_{SW}: length of returned Status Words
#3-#4	2	‘XXXXh’	Processing Status of the unprotected response APDU

#5	1	‘8Eh’	T_{CC}: Tag for cryptographic checksum
#6	1	‘XXh’	L_{CC}: Length of following cryptographic checksum ‘04h’ for Generation 1 secure messaging (see Appendix 11 Part A) ‘08h’, ‘0Ch’ or ‘10h’ depending on AES key length for Generation 2 secure messaging (see Appendix 11 Part B)
#7-#(6+L)	L	‘XX..XXh’	Cryptographic checksum
SW	2	‘XXXXh’	Status Words (SW1,SW2)

The ‘regular’ processing states, described for the UPDATE BINARY command with no secure messaging (see §3.5.3.1), can be returned using the response message structure described above.

Additionally, some errors specifically related to secure messaging can happen. In that case, the processing state is simply returned, with no secure messaging structure involved:

TCS_60 **Response Message if error in secure messaging**

Byte	Length	Value	Description
SW	2	‘XXXXh’	Status Words (SW1,SW2)

- If no current session key is available, the processing state ‘**6A88**’ is returned.
- If some expected data objects (as specified above) are missing in the secure messaging format, the processing state ‘**6987**’ is returned: this error happens if an expected tag is missing or if the command body is not properly constructed.
- If some data objects are incorrect, the processing state returned is ‘**6988**’: this error happens if all the required tags are present but some lengths are different from the ones expected.

— If the verification of the cryptographic checksum fails, the processing state returned is '**6688**'.

3.5.3.2 *Command with short EF identifier*

This command variant enables the IFD to select an EF by means of a short EF identifier and write data from this EF.

TCS_61 A tachograph card shall support this command variant for all Elementary Files with a specified short EF identifier. These short EF identifiers are specified in chapter 4.

TCS_62 **Command Message**

Byte	**Length**	**Value**	**Description**
CLA	1	'00h'	
INS	1	'D6h'	Update Binary
P1	1	'XXh'	Bit 8 is set to 1 Bit 7 and 6 are set to 00 Bit 5 — 1 encode the short EF identifier of the corresponding EF
P2	1	'XXh'	Encodes an offset from 0 to 255 bytes in the EF referenced by P1
Lc	1	'NNh'	Lc Length of data to Update. Number of bytes to be written.
#6-#(5+NN)	NN	'XX..XXh'	Data to be written

TCS_63 **Response Message**

Byte	**Length**	**Value**	**Description**
SW	2	'XXXXh'	Status Words (SW1,SW2)

Note: The short EF identifiers used for the generation 2 tachograph application are specified in chapter 4.

If P1 encodes a short EF identifier and the command is successful, the identified EF becomes the currently selected EF (current EF).

— If the command is successful, the card returns '**9000**'.

— If the file corresponding with the short EF identifier is not found, the processing state returned is '**6A82**'.

— If the security conditions of the selected file are not satisfied, the command is interrupted with '**6982**'.

— If the Offset is not compatible with the size of the EF (Offset > EF size), the processing state returned is '**6B00**'.

— If the size of the data to be written is not compatible with the size of the EF (Offset + Lc > EF size) the processing state returned is ‘**6700**’.

— [F2If an integrity error is detected within the file attributes, the card shall consider the file as corrupted and unrecoverable, the processing state returned is ‘**6400**’ or ‘**6500**’.]

— If writing is unsuccessful, the processing state returned is ‘**6581**’.

3.5.3.3 *Command with odd instruction byte*

This command variant enables the IFD to write data to an EF with 32 768 bytes or more.

TCS_64 A tachograph card which supports EFs with 32 768 bytes or more shall support this command variant for these EFs. A tachograph card may or may not support this command variant for other EFs.

TCS_65 **Command Message**

Byte	Length	Value	Description
CLA	1	‘00h’	
INS	1	‘D7h’	Update Binary
P1	1	‘00h’	Current EF
P2	1	‘00h’	
Lc	1	‘NNh’	Lc Length of data in the command data field
#6-#(5+NN)	NN	‘XX..XXh’	Offset data object with tag ‘54h’ || Discretionary data object with tag ‘53h’ that encapsulates the data to be written

The IFD shall encode the offset data object's and the discretionary data object's length with the minimum possible number of octets, i.e. using the length byte ‘01h’ the IFD shall encode an offset / length from 0 to 255 and using the length byte ‘02h’ an offset / length from ‘256’ up to ‘65 535’ bytes.

TCS_66 **Response Message**

Byte	Length	Value	Description
SW	2	‘XXXXh’	Status Words (SW1,SW2)

— If the command is successful, the card returns ‘**9000**’.

— If no EF is selected, the processing state returned is ‘**6986**’.

— If the security conditions of the selected file are not satisfied, the command is interrupted with ‘**6982**’.

— If the Offset is not compatible with the size of the EF (Offset > EF size), the processing state returned is ‘**6B00**’.

— If the size of the data to be written is not compatible with the size of the EF (Offset + Lc > EF size) the processing state returned is '**6700**'.

— If an integrity error is detected within the file attributes, the card shall consider the file as corrupted and unrecoverable, the processing state returned is '**6400**' or '**6500**'.

— If writing is unsuccessful, the processing state returned is '**6581**'.

3.5.3.3.1 *Command with secure messaging (example)*

The following example illustrates the usage of secure messaging if the security condition SM-MAC-G2 applies.

TCS_67 Command message

Byte	Length	Value	Description
CLA	1	'0Ch'	Secure Messaging asked
INS	1	'D7h'	Update Binary
P1	1	'00h'	Current EF
P2	1	'00h'	
Lc	1	'XXh'	Length of the secured data field
#6	1	'B3h'	Tag for plain value data encoded in BER-TLV
#7	L	'NNh' or '81 NNh'	L_{PV}: length of transmitted data. L is 2 bytes if L_{PV} > 127 bytes.
#(7+L)-#(6+L+NN)	NN	'XX..XXh'	Plain Data encoded in BER-TLV, i.e. offset data object with tag '54h' || Discretionary data object with tag '53h' that encapsulates the data to be written
#(7+L+NN)	1	'8Eh'	T_{CC}: Tag for cryptographic checksum
#(8+L+NN)	1	'XXh'	L_{CC}: Length of following cryptographic checksum '08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging

			(see Appendix 11 Part B)
#(9+L+NN)-#(8+M +L+NN)	M	'XX..XXh'	Cryptographic checksum
Le	1	'00h'	As specified in ISO/ IEC 7816-4

TCS_68 Response message if the command is successful

Byte	Length	Value	Description
#1	1	'99h'	T_{SW}: Tag for Status Words (to be protected by CC)
#2	1	'02h'	L_{SW}: length of returned Status Words
#3-#4	2	'XXXXh'	Processing Status of the unprotected response APDU
#5	1	'8Eh'	T_{CC}: Tag for cryptographic checksum
#6	1	'XXh'	L_{CC}: Length of following cryptographic checksum '08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix 11 Part B)
#7-#(6+L)	L	'XX..XXh'	Cryptographic checksum
SW	2	'XXXXh'	Status Words (SW1,SW2)

3.5.4 *GET CHALLENGE*

This command is compliant with ISO/IEC 7816-4, but has a restricted usage compared to the command defined in the norm.

The GET CHALLENGE command asks the card to issue a challenge in order to use it in a security related procedure in which a cryptogram or some ciphered data are sent to the card.

TCS_69 The Challenge issued by the card is only valid for the next command, which uses a challenge, sent to the card.

TCS_70 **Command Message**

Byte	Length	Value	Description
CLA	1	‘00h’	
INS	1	‘84h’	INS
P1	1	‘00h’	P1
P2	1	‘00h’	P2
Le	1	‘08h’	Le (Length of Challenge expected).

TCS_71 **Response Message**

Byte	Length	Value	Description
#1-#8	8	‘XX..XXh’	Challenge
SW	2	‘XXXXh’	Status Words (SW1,SW2)

— If the command is successful, the card returns ‘**9000**’.

— If Le is different from ‘08h’, the processing state is ‘**6700**’.

— If parameters P1-P2 are incorrect, the processing state is ‘**6A86**’.

3.5.5 *VERIFY*

This command is compliant with ISO/IEC 7816-4, but has a restricted usage compared to the command defined in the norm.

Only the workshop card is required to support this command.

Other types of tachograph cards may or may not implement this command, but for these cards no reference CHV is personalized. Therefore these cards cannot perform this commend successfully. For other types of tachograph cards than workshop cards the behavior, i.e. the error code returned, is out of the scope of this specification, if this command is sent.

The Verify command initiates the comparison in the card of the CHV (PIN) data sent from the command with the reference CHV stored in the card.

[F2TCS_72 The PIN entered by the user must be ASCII encoded and right padded with ‘FFh’ bytes up to a length of 8 bytes by the IFD, see also the data type WorkshopCardPIN in Appendix 1.**]**

TCS_73 The tachograph applications generation 1 and 2 shall use the same reference CHV.

TCS_74 The tachograph card shall check whether the command is encoded correctly. If the command is not encoded correctly the card shall not compare the CHV values, not decrement the remaining CHV attempt counter and not reset the security status ‘PIN_Verified’, but abort the command. A command is encoded correctly, if the CLA, INS, P1, P2, Lc bytes have the specified values, Le is absent, and the command data field has the correct length.

TCS_75 If the command is successful, the remaining CHV attempt counter is reinitialised. The initial value of the remaining CHV attempt counter is 5. If the command is successful the card shall set the internal security status 'PIN_Verified'. The card shall reset this security status, if the card is reset or if the CHV code transmitted in the command does not match the stored reference CHV.

Note: Using the same reference CHV and a global security status prevents that a workshop employee must re-enter the PIN after a selection of another tachograph application DF.

TCS_76 An unsuccessful comparison is recorded in the card, i.e. the remaining CHV attempts counter shall be decremented by one, in order to limit the number of further attempts of the use of the reference CHV.

TCS_77 **Command Message**

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'20h'	INS
P1	1	'00h'	P1
P2	1	'00h'	P2 (the verified CHV is implicitly known)
Lc	1	'08h'	Length of CHV code transmitted
#6-#13	8	'XX..XXh'	CHV

TCS_78 **Response Message**

Byte	Length	Value	Description
SW	2	'XXXXh'	Status Words (SW1,SW2)

— If the command is successful, the card returns '**9000**'.

— If the reference CHV is not found, the processing state returned is '**6A88**'.

— If the CHV is blocked, (the remaining attempt counter of the CHV is null), the processing state returned is '**6983**'. Once in that state, the CHV can never be successfully presented anymore.

— If the comparison is unsuccessful, the remaining attempt Counter is decreased and the status '**63CX**' is returned (X>0 and X equals the remaining CHV attempts counter.

— If the reference CHV is considered corrupted, the processing state returned is '**6400**' or '**6581**'.

— If Lc is different from '08h', the processing state is '**6700**'.

3.5.6 *GET RESPONSE*

This command is compliant with ISO/IEC 7816-4.

This command (only necessary and available for T=0 Protocol) is used to transmit prepared data from the card to the interface device (case where a command had included both Lc and Le).

The GET RESPONSE command has to be issued immediately after the command preparing the data, otherwise, the data are lost. After the execution of the GET RESPONSE command (except if the error '**61xx**' or '**6Cxx**' occur, see below), the previously prepared data are no longer available.

TCS_79 **Command Message**

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'C0h'	
P1	1	'00h'	
P2	1	'00h'	
Le	1	'XXh'	Number of bytes expected

TCS_80 **Response Message**

Byte	Length	Value	Description
#1-#X	X	'XX..XXh'	Data
SW	2	'XXXXh'	Status Words (SW1,SW2)

— If the command is successful, the card returns '**9000**'.

— If no data have been prepared by the card, the processing state returned is '**6900**' or '**6F00**'.

— If Le exceeds the number of available bytes or if Le is null, the processing state returned is '**6Cxx**', where xx denotes the exact number of available bytes. In that case, the prepared data are still available for a subsequent GET RESPONSE command.

— If Le is not null and is smaller than the number of available bytes, the required data are sent normally by the card, and the processing state returned is '**61xx**', where 'xx' indicates a number of extra bytes still available by a subsequent GET RESPONSE command.

— If the command is not supported (protocol T=1), the card returns '**6D00**'.

3.5.7 *PSO: VERIFY CERTIFICATE*

This command is compliant with ISO/IEC 7816-8, but has a restricted usage compared to the command defined in the norm.

The VERIFY CERTIFICATE command is used by the card to obtain a Public Key from the outside and to check its validity.

3.5.7.1 *Generation 1 Command — Response pair*

TCS_81 This command variant is only supported by a generation 1 tachograph application.

TCS_82 When a VERIFY CERTIFICATE command is successful, the Public Key is stored for a future use in the Security environment. This key shall be explicitly set for the use in security related commands (INTERNAL AUTHENTICATE, EXTERNAL

AUTHENTICATE or VERIFY CERTIFICATE) by the MSE command (see § 3.5.11) using its key identifier.

TCS_83 In any case, the VERIFY CERTIFICATE command uses the public key previously selected by the MSE command to open the certificate. This public key must be the one of a Member State or of Europe.

TCS_84 **Command Message**

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'2Ah'	Perform Security Operation
P1	1	'00h'	P1
P2	1	'AEh'	P2: non BER-TLV coded data (concatenation of data elements)
Lc	1	'C2h'	Lc: Length of the certificate, 194 bytes
#6-#199	194	'XX..XXh'	Certificate: concatenation of data elements (as described in Appendix 11)

TCS_85 **Response Message**

Byte	Length	Value	Description
SW	2	'XXXXh'	Status Words (SW1,SW2)

— If the command is successful, the card returns '**9000**'.

— If the certificate verification fails, the processing state returned is '**6688**'. The verification and unwrapping process of the certificate is described in Appendix 11 for G1 and G2.

— If no Public Key is present in the Security Environment, '**6A88**' is returned.

— If the selected public key (used to unwrap the certificate) is considered corrupted, the processing state returned is '**6400**' or '**6581**'.

— Generation 1 only: If the selected public key (used to unwrap the certificate) has a CHA.LSB (`CertificateHolderAuthorisation.equipmentType`) different from '00' (i.e. is not the one of a Member State or of Europe), the processing state returned is '**6985**'.

3.5.7.2 *Generation 2 Command — Response pair*

Depending on the curve size ECC certificates may be so long that they cannot be transmitted in a single APDU. In this case command chaining according to ISO/IEC 7816-4 must be applied and the certificate transmitted in two consecutive PSO: Verify Certificate APDUs.

The certificate structure and the domain parameters are defined in Appendix 11.

TCS_86 The command can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_33.

TCS_87 **Command Message**

Byte	**Length**	**Value**	**Description**
CLA	1	‘X0h’	CLA byte indicating command chaining: ‘00h’ the only or last command of the chain ‘10h’ not the last command of a chain
INS	1	‘2Ah’	Perform Security Operation
P1	1	‘00h’	
P2	1	‘BEh’	Verify self-descriptive certificate
Lc	1	‘XXh’	Length of the command data field, see TCS_88 and TCS_89.
#6-#5+L	L	‘XX..XXh’	DER-TLV encoded data: ECC Certificate Body data object as first data object concatenated with the ECC Certificate Signature data object as second data object or a part of this concatenation. The tag ‘7F21’ and the corresponding length shall not be transmitted. The order of these data objects is fixed.

TCS_88 For short length APDUs the following provisions apply: The IFD shall use the minimum number of APDUs required to transmit the command payload and transmit the maximum number of bytes in the first command APDU according to the value of the Information Field Size Card Byte, see TCS_14. If the IFD behaves differently, the behavior of the card is out of scope.

TCS_89 For extended length APDUs the following provisions apply: If the certificate does not fit into a single APDU, the card shall support command chaining. The IFD shall use the minimum number of APDUs required to transmit the command payload and transmit the maximum number of bytes in the first command APDU. If the IFD behaves differently, the behavior of the card is out of scope.

Note: According to Appendix 11 the card stores the certificate or the relevant contents of the certificate and updates its currentAuthenticatedTime.

The response message structure and status words are as defined in TCS_85.

TCS_90 In addition to the error codes listed in TCS_85, the card may return the following error codes:

— If the selected public key (used to unwrap the certificate) has a CHA.LSB (CertificateHolderAuthorisation.equipmentType) that is not suitable for the certificate verification according to Appendix 11, the processing state returned is '**6985**'.

— If the currentAuthenticatedTime of the card is later than the Certificate Expiration Date, the processing state returned is '**6985**'.

— If the last command of the chain is expected, the card returns '**6883**'.

— If incorrect parameters are sent in the command data field, the card returns '**6A80**' (also used in case the data objects are not sent in the specified order).

3.5.8 *INTERNAL AUTHENTICATE*

This command is compliant with ISO/IEC 7816-4.

TCS_91 All tachograph cards shall support this command in the DF Tachograph generation 1. The command may or may not be accessible in the MF and / or the DF Tachograph_G2. If so, the command shall terminate with a suitable error code as the private key of the card (Card.SK) for the generation 1 authentication protocol is only accessible in the DF_Tachograph generation 1.

Using the INTERNAL AUTHENTICATE command, the IFD can authenticate the card. The authentication process is described in Appendix 11. It includes the following statements:

TCS_92 The INTERNAL AUTHENTICATE command uses the card Private Key (implicitly selected) to sign authentication data including K1 (first element for session key agreement) and RND1, and uses the Public Key currently selected (through the last MSE command) to encrypt the signature and form the authentication token (more details in Appendix 11).

TCS_93 **Command Message**

Byte	**Length**	**Value**	**Description**
CLA	1	'00h'	CLA
INS	1	'88h'	INS
P1	1	'00h'	P1
P2	1	'00h'	P2
Lc	1	'10h'	Length of data sent to the card

#6 — #13	8	'XX..XXh'	Challenge used to authenticate the card
#14 -#21	8	'XX..XXh'	VU.CHR (see Appendix 11)
Le	1	'80h'	Length of the data expected from the card

TCS_94 **Response Message**

Byte	Length	Value	Description
#1-#128	128	'XX..XXh'	Card authentication token (see Appendix 11)
SW	2	'XXXXh'	Status Words (SW1,SW2)

— If the command is successful, the card returns '**9000**'.

— If no Public Key is present in the Security Environment, the processing state returned is '**6A88**'.

— If no Private Key is present in the Security Environment, the processing state returned is '**6A88**'.

— If VU.CHR does not match the current public key identifier, the processing state returned is '**6A88**'.

— If the selected private key is considered corrupted, the processing state returned is '**6400**' or '**6581**'.

[F2TCS_95 If the INTERNAL AUTHENTICATE command is successful, the current generation 1 session key, if existing, is erased and no longer available. In order to have a new generation 1 session key available, the EXTERNAL AUTHENTICATE command for the generation 1 authentication mechanism must be successfully performed.

Note: For generation 2 session keys see Appendix 11 CSM_193 and CSM_195. If generation 2 session keys are established and the tachograph card receives the plain INTERNAL AUTHENTICATE command APDU, it aborts the generation 2 secure messaging session and destroys the generation 2 session keys.]

3.5.9 *EXTERNAL AUTHENTICATE*

This command is compliant with ISO/IEC 7816-4.

Using the EXTERNAL AUTHENTICATE command, the card can authenticate the IFD. The authentication process is described in Appendix 11 for Tachograph G1 and G2 (VU authentication).

TCS_96 The command variant for the generation 1 mutual authentication mechanism is only supported by a generation 1 tachograph application.

[F2TCS_97 The command variant for the second generation VU-card mutual authentication can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_34. If this generation 2 EXTERNAL AUTHENTICATE command is successful, the current generation 1 session key, if existing, is erased and no longer available.

Note: For generation 2 session keys see Appendix 11 CSM_193 and CSM_195. If generation 2 session keys are established and the tachograph card receives the plain EXTERNAL AUTHENTICATE command APDU, it aborts the generation 2 secure messaging session and destroys the generation 2 session keys.]

TCS_98 **Command Message**

Byte	Length	Value	Description
CLA	1	‘00h’	CLA
INS	1	‘82h’	INS
P1	1	‘00h’	Keys and algorithms implicitly known
P2	1	‘00h’	
Lc	1	‘XXh’	Lc (Length of the data sent to the card)
#6-#(5+L)	L	‘XX..XXh’	Generation 1 authentication: Cryptogram (see Appendix 11 Part A) Generation 2 authentication: Signature generated by the IFD (see Appendix 11 Part B)

TCS_99 **Response Message**

Byte	Length	Value	Description
SW	2	‘XXXXh’	Status Words (SW1,SW2)

- If the command is successful, the card returns ‘**9000**’.
- If the CHA of the currently set public key is not the concatenation of the Tachograph application AID and of a VU equipment Type, the processing state returned is ‘**6F00**’.
- If the command is not immediately preceded with a GET CHALLENGE command, the processing state returned is ‘**6985**’.

The Generation 1 Tachograph application may return the following additional error codes:

- If no Public Key is present in the Security Environment, ‘**6A88**’ is returned.
- If no Private Key is present in the Security Environment, the processing state returned is ‘**6A88**’.
- If the verification of the cryptogram is wrong, the processing state returned is ‘**6688**’.
- If the selected private key is considered corrupted, the processing state returned is ‘**6400**’ or ‘**6581**’.

The command variant for the Generation 2 authentication may return the following additional error code:

— If signature verification failed, the card returns '**6300**'.

3.5.10 *GENERAL AUTHENTICATE*

This command is used for the generation 2 chip authentication protocol specified in Appendix 11 Part B and is compliant with ISO/IEC 7816-4.

TCS_100The command can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_34.

TCS_101**Command Message**

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'86h'	
P1	1	'00h'	Keys and protocol implicitly known
P2	1	'00h'	
Lc	1	'NNh'	Lc: length of subsequent data field
#6-#(5+L)	L	'7Ch' + L_{7C} + '80h' + L_{80} + 'XX..XXh'	DER-TLV encoded ephemeral public key value (see Appendix 11) The VU shall send the data objects in this order.
[F1 5 + L + 1	1	'00h'	As specified in ISO/IEC 7816-4**]**

TCS_102**Response Message**

Byte	Length	Value	Description
#1-#L	L	'7Ch' + L_{7C} + '81h' + '08h' + 'XX..XXh' + '82h' + L_{82} + 'XX..XXh'	DER-TLV encoded Dynamic Authentication Data: nonce and authentication token (see Appendix 11)
SW	2	'XXXXh'	Status Words (SW1,SW2)

— If the command is successful, the card returns '**9000**'.

— The card returns '**6A80**' to indicate incorrect parameters in data field.

— The card returns '**6982**' if the External Authenticate command has not been performed successfully

The response Dynamic Authentication Data object ‘7Ch’

— must be present if the operation is successful, i.e. the Status Words are ‘**9000**’,

— must be absent in case of an execution error or checking error, i.e. if the Status Words are in the range ‘**6400**’ — ‘**6FFF**’, and

— may be absent in case of a warning, i.e. if the Status Words are in the range ‘**6200**’ — ‘**63FF**’.

3.5.11 *MANAGE SECURITY ENVIRONMENT*

This command is used to set a public key for authentication purpose.

3.5.11.1 *Generation 1 Command — Response pair*

This command is compliant with ISO/IEC 7816-4. The use of this command is restricted regarding the related standard.

TCS_103This command is only supported by a generation 1 tachograph application.

TCS_104The key referenced in the MSE data field remains the current public key until the next correct MSE command, a DF is selected or the card is reset.

TCS_105If the key referenced is not (already) present into the card, the security environment remains unchanged.

TCS_106**Command Message**

Byte	**Length**	**Value**	**Description**
CLA	1	‘00h’	CLA
INS	1	‘22h’	INS
P1	1	‘C1h’	P1: referenced key valid for all cryptographic operations
P2	1	‘B6h’	P2 (referenced data concerning Digital Signature)
Lc	1	‘0Ah’	Lc: length of subsequent data field
#6	1	‘83h’	Tag for referencing a public key in asymmetric cases
#7	1	‘08h’	Length of the key reference (key identifier)
#8-#15	8	‘XX..XXh’	Key identifier as specified in Appendix 11

TCS_107**Response Message**

Byte	Length	Value	Description
SW	2	'XXXXh'	Status Words (SW1,SW2)

- — If the command is successful, the card returns '**9000**'.
- — If the referenced key is not present into the card, the processing state returned is '**6A88**'.
- — If some expected data objects are missing in the secure messaging format, the processing state '**6987**' is returned. This can happen if the tag '83h' is missing.
- — If some data objects are incorrect, the processing state returned is '**6988**'. This can happen if the length of the key identifier is not '08h'.
- — If the selected key is considered corrupted, the processing state returned is '**6400**' or '**6581**'.

3.5.11.2 *Generation 2 Command — Response pairs*

For the Generation 2 authentication the tachograph card supports the following MSE: Set command versions which are compliant with ISO/IEC 7816-4. These command versions are not supported for the Generation 1 authentication.

3.5.11.2.1*MSE:SET AT for Chip Authentication*

The following MSE:SET AT command is used to select the parameters for the Chip Authentication that is performed by a subsequent General Authenticate command.

TCS_108The command can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_34.

TCS_109**MSE:SET AT Command Message for Chip Authentication**

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'22h'	
P1	1	'41h'	Set for internal authentication
P2	1	'A4h'	Authentication
Lc	1	'NNh'	Lc: length of subsequent data field
#6-#(5+L)	L	'80h' + '0Ah' + 'XX..XXh'	DER-TLV encoded cryptographic mechanism reference: Object Identifier of Chip Authentication (value only, Tag '06h' is omitted). See Appendix 1 for the values of object identifiers; the byte notation shall be used. See Appendix

			11 for guidance on how to select one of these object identifiers.

3.5.11.2.2*MSE:SET AT for VU Authentication*

The following MSE:SET AT command is used to select the parameters and keys for the VU Authentication that is performed by a subsequent External Authenticate command.

TCS_110The command can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_34.

TCS_111**MSE:SET AT Command Message for VU Authentication**

Byte	Length	Value	Description
CLA	1	‘00h’	
INS	1	‘22h’	
P1	1	‘81h’	Set for external authentication
P2	1	‘A4h’	Authentication
Lc	1	‘NNh’	Lc: length of subsequent data field
#6-#(5+L)	L	‘80h’ + ‘0Ah’ + ‘XX..XXh’	DER-TLV encoded cryptographic mechanism reference: Object Identifier of VU Authentication (value only, Tag ‘06h’ is omitted). See Appendix 1 for the values of object identifiers; the byte notation shall be used. See Appendix 11 for guidance on how to select one of these object identifiers.
		‘83h’ + ‘08h’ + ‘XX..XXh’	DER-TLV encoded reference of the VU public key by the Certificate Holder Reference mentioned in its certificate.
		‘91h’ + L_{91} + ‘XX..XXh’	DER-TLV encoded compressed representation of the ephemeral public key

			of the VU that will be used during Chip Authentication (see Appendix 11)

3.5.11.2.3*MSE:SET DST*

The following MSE:SET DST command is used to set a public key either

- for the verification of a signature that is provided in a subsequent PSO: Verify Digital Signature command or
- for the signature verification of a certificate that is provided in a subsequent PSO: Verify Certificate command

TCS_112The command can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_33.

TCS_113**MSE:SET DST Command Message**

Byte	**Length**	**Value**	**Description**
CLA	1	‘00h’	
INS	1	‘22h’	
P1	1	‘81h’	Set for verification
P2	1	‘B6h’	Digital Signature
Lc	1	‘NNh’	Lc: length of subsequent data field
#6-#(5+L)	L	‘83h’ + ‘08h’ + ‘XX...XXh’	DER-TLV encoded reference of a public key, i.e. the Certificate Holder Reference in the certificate of the public key (see Appendix 11)

For all command versions the response message structure and status words are given by:

TCS_114**Response Message**

Byte	**Length**	**Value**	**Description**
SW	2	‘XXXXh’	Status Words (SW1,SW2)

- If the command is successful, the card returns ‘**9000**’. The protocol has been selected and initialised.
- ‘**6A80**’ indicates incorrect parameters in the command data field.
- ‘**6A88**’ indicates that referenced data (i.e. a referenced key) is not available.
- [F1If the currentAuthenticatedTime of the card is later than the Expiration Date of the selected public key, the processing state returned is **‘6A88’.**

Note: In the case of a MSE: SET AT for VU Authentication command, the referenced key is a VU_MA public key. The card shall set the VU_MA public key for use, if available in its memory, which matches the Certificate Holder Reference (CHR) given in the command data field (the card can identify VU_MA public keys by means of the certificate's CHA field). A card shall return '6A 88' to this command in case only the VU_Sign public key or no public key of the Vehicle Unit is available. See the definition of the CHA field in Appendix 11 and of data type equipmentType in Appendix 1.

Similarly, in case an MSE: SET DST command referencing an EQT (i.e. a VU or a card) is sent to a control card, according to CSM_234 the referenced key is always an EQT_Sign key that has to be used for the verification of a digital signature. According to Figure 13 in Appendix 11, the control card will always have stored the relevant EQT_Sign public key. In some cases, the control card may have stored the corresponding EQT_MA public key. The control card shall always set the EQT_Sign public key for use when it receives an MSE: SET DST command.**]**

3.5.12 *PSO: HASH*

This command is used to transfer to the card the result of a hash calculation on some data. This command is used for the verification of digital signatures. The hash value is stored temporarily for the subsequent command PSO: Verify Digital Signature

This command is compliant with ISO/IEC 7816-8. The use of this command is restricted regarding the related standard.

Only the control card is required to support this command in the DF Tachograph and DF Tachograph_G2.

Other types of tachograph cards may or may not implement this command. The command may or may not be accessible in the MF.

The control card application generation 1 supports only SHA-1.

TCS_115The temporarily stored hash value shall be deleted if a new hash value is computed by means of the PSO: HASH command, if a DF is selected, and if the tachograph card is reset.

TCS_116**Command Message**

Byte	Length	Value	Description
CLA	1	'00h'	CLA
INS	1	'2Ah'	Perform Security Operation
P1	1	'90h'	Return Hash code
P2	1	'A0h'	Tag: data field contains DOs relevant for hashing
Lc	1	'XXh'	Length Lc of the subsequent data field
#6	1	'90h'	Tag for the hash code
#7	1	'XXh'	Length L of the hash code:

			'14h' in Generation 1 application (see Appendix 11 Part A) '20h', '30h' or '40h' in Generation 2 application (see Appendix 11 Part B)
#8-#(7+L)	L	'XX..XXh'	Hash code

TCS_117**Response Message**

Byte	**Length**	**Value**	**Description**
SW	2	'XXXXh'	Status Words (SW1,SW2)

— If the command is successful, the card returns '**9000**'.

— If some expected data objects (as specified above) are missing, the processing state '**6987**' is returned. This can happen if one of the tag '90h' is missing.

— If some data objects are incorrect, the processing state returned is '**6988**'. This error happens if the required tag is present but with a length different from '14h' for SHA-1, '20h' for SHA-256, '30h' for SHA-384, '40h' for SHA-512 (Generation 2 application).

3.5.13 *PERFORM HASH of FILE*

This command is not compliant with ISO/IEC 7816-8. Thus the CLA byte of this command indicates that there is a proprietary use of the PERFORM SECURITY OPERATION / HASH.

Only the driver card and the workshop card are required to support this command in the DF Tachograph and DF Tachograph_G2.

Other types of tachograph cards may or may not implement this command. If a company or control card implements this command, the command shall be implemented as specified in this chapter.

The command may or may not be accessible in the MF. If so, the command shall be implemented as specified in this chapter, i.e. shall not allow the calculation of a hash value, but terminate with a suitable error code.

TCS_118The PERFORM HASH of FILE command is used to hash the data area of the currently selected transparent EF.

TCS_119A tachograph card shall support this command only for the EFs that are listed in chapter 4 under the DF_Tachograph and DF_Tachograph_G2 with the following exception. A tachograph card shall not support the command for the EF Sensor_Installation_Data of DF Tachograph_G2..

TCS_120 The result of the hash operation is stored temporarily in the card. It can then be used to get a digital signature of the file, using the PSO: COMPUTE DIGITAL SIGNATURE command.

[F2TCS_121 The temporarily stored hash of file value shall be deleted if a new hash of file value is computed by means of the PERFORM HASH of FILE command, if a DF is selected, and if the tachograph card is reset.]

TCS_122 The Tachograph Generation 1 application shall support SHA-1.

[F2TCS_123 The Tachograph Generation 2 application shall support the SHA-2 algorithm (SHA-256, SHA-384 or SHA-512), specified by the cipher suite in Appendix 11 Part B for the card signature key Card_Sign.]

TCS_124 **Command Message**

[F2**Byte**	**Length**	**Value**	**Description**
CLA	1	‘80h’	CLA
INS	1	‘2Ah’	Perform Security Operation
P1	1	‘90h’	Tag: Hash
P2	1	‘00h’	Algorithm implicitly known For the Tachograph Generation 1 application: SHA-1 For the Tachograph Generation 2 application: SHA-2 algorithm (SHA-256, SHA-384 or SHA-512) defined by the cipher suite in Appendix 11 Part B for the card signature key Card_Sign]

TCS_125 **Response Message**

Byte	**Length**	**Value**	**Description**
SW	2	‘XXXXh’	Status Words (SW1,SW2)

— If the command is successful, the card returns ‘**9000**’.

— If the current EF does not allow this command (EF Sensor_Installation_Data in DF Tachograph_G2), the processing state ‘**6985**’ is returned.

— If the selected EF is considered corrupted (file attributes or stored data integrity errors), the processing state returned is '**6400**' or '**6581**'.

— If the selected file is not a transparent file or if there is no current EF, the processing state returned is '**6986**'.

3.5.14 *PSO: COMPUTE DIGITAL SIGNATURE*

[F2This command is used to compute the digital signature of previously computed hash code (see PERFORM HASH of FILE, §3.5.13).

Only the driver card and the workshop card are required to support this command in the DF Tachograph and DF Tachograph_G2.

Other types of tachograph cards may or may not implement this command. In case of the Generation 2 tachograph application, only the driver card and the workshop card have a generation 2 signature key, other cards are not able to successfully perform the command and terminate with a suitable error code.

The command may or may not be accessible in the MF. If the command is not accessible in the MF, it shall terminate with a suitable error code.

This command is compliant with ISO/IEC 7816-8. The use of this command is restricted regarding the related standard.**]**

TCS_126This command shall not compute a digital signature of previously computed hash code with the PSO: HASH command.

TCS_127The card private key is used to compute the digital signature and is implicitly known by the card.

TCS_128The Generation 1 tachograph application performs a digital signature using a padding method compliant with PKCS1 (see Appendix 11 for details).

TCS_129The Generation 2 tachograph application computes an elliptic curve based digital signature (see Appendix 11 for details).

TCS_130**Command Message**

Byte	Length	Value	Description
CLA	1	'00h'	CLA
INS	1	'2Ah'	Perform Security Operation
P1	1	'9Eh'	Digital signature to be returned
P2	1	'9Ah'	Tag: data field contains data to be signed. As no data field is included, the data are supposed to be already present in the card (hash of file)
Le	1	'NNh'	Length of the expected signature

TCS_131**Response Message**

Byte	**Length**	**Value**	**Description**
#1-#L	L	'XX..XXh'	Signature of the previously computed hash
SW	2	'XXXXh'	Status Words (SW1,SW2)

— If the command is successful, the card returns '**9000**'.

— If the implicitly selected private key is considered as corrupted, the processing state returned is '**6400**' or '**6581**'.

— If the hash which was computed in a previous Perform Hash of File command is not available, the processing state returned is '**6985**'.

3.5.15 *PSO: VERIFY DIGITAL SIGNATURE*

This command is used to verify the digital signature, provided as an input, whose hash is known to the card. The signature algorithm is implicitly known by the card.

This command is compliant with ISO/IEC 7816-8. The use of this command is restricted regarding the related standard.

Only the control card is required to support this command in the DF Tachograph and DF Tachograph_G2.

Other types of tachograph cards may or may not implement this command. The command may or may not be accessible in the MF.

TCS_132The VERIFY DIGITAL SIGNATURE command always uses the public key selected by the previous Manage Security Environment MSE: Set DST command and the previous hash code entered by a PSO: HASH command.

TCS_133**Command Message**

[F2**Byte**	**Length**	**Value**	**Description**
CLA	1	'00h'	CLA
INS	1	'2Ah'	Perform Security Operation
P1	1	'00h'	
P2	1	'A8h'	Tag: data field contains DOs relevant for verification
Lc	1	'XXh'	Length Lc of the subsequent data field
#6	1	'9Eh'	Tag for Digital Signature

#7 or #7-#8	L	'NNh' or '81 NNh'	Length of digital signature (L is 2 bytes if the digital signature is longer than 127 bytes): 128 bytes coded in accordance with Appendix 11 Part A for Tachograph Generation 1 application. Depending on the selected curve for Tachograph Generation 2 application (see Appendix 11 Part B).
#(7+L)-#(6+L+NN)	NN	'XX..XXh'	Digital signature content**]**

TCS_134**Response Message**

Byte	Length	Value	Description
SW	2	'XXXXh'	Status Words (SW1,SW2)

- If the command is successful, the card returns '**9000**'.
- If the verification of the signature fails, the processing state returned is '**6688**'. The verification process is described in Appendix 11.
- If no public key is selected, the processing state returned is '**6A88**'.
- If some expected data objects (as specified above) are missing, the processing state '**6987**' is returned. This can happen if one of the required tag is missing.
- If no hash code is available to process the command (as a result of a previous PSO: Hash command), the processing state returned is '**6985**'.
- If some data objects are incorrect, the processing state returned is '**6988**'. This can happen if one of the required data objects length is incorrect.
- If the selected public key is considered corrupted, the processing state returned is '**6400**' or '**6581**'.

— [F1If the selected public key (used to verify the digital signature) has a CHA.LSB (CertificateHolderAuthorisation.equipmentType) that is not suitable for the digital signature verification according to Appendix 11, the processing state returned is '**6985**'.]

3.5.16 *PROCESS DSRC MESSAGE*

This command is used to verify the integrity and authenticity of the DSRC message and to decipher the data communicated from a VU to a control authority or a workshop over the DSRC link. The card derives the encryption key and the MAC key used to secure the DSRC message as described in Appendix 11 Part B chapter 13.

Only the control card and the workshop card are required to support this command in the DF Tachograph_G2.

Other types of tachograph cards may or may not implement this command, but shall not have a DSRC master key. Therefore these cards cannot perform the command successfully, but terminate with a suitable error code.

The command may or may not be accessible in the MF and / or the DF Tachograph. If so, the command shall terminate with a suitable error code.

TCS_135The DSRC master key is accessible only in the DF Tachograph_G2, i.e. the control and workshop card shall support a successful execution of the command only in the DF Tachograph_G2.

TCS_136The command shall only decrypt the DSRC data and verify the cryptographic checksum, but not interpret the input data.

TCS_137The order of the data objects in the command data field is fixed by this specification.

TCS_138**Command Message**

Byte	**Length**	**Value**	**Description**
CLA	1	'80h'	Proprietary CLA
INS	1	'2Ah'	Perform Security Operation
P1	1	'80h'	Response data: plain value
P2	1	'B0h'	Command data: plain value encoded in BER-TLV and including SM DOs
Lc	1	'NNh'	Length Lc of the subsequent data field
#6-#(5+L)	L	'87h' + L_{87} + 'XX..XXh'	DER-TLV encoded padding-content indicator byte followed by encrypted tachograph payload. For the padding-content indicator byte the

			value ‘00h’ (‘no further indication’ according to ISO/IEC 7816-4:2013 Table 52) shall be used. For the encryption mechanism see Appendix 11, Part B chapter 13. Allowed values for the length L_{87} are the multiples of the AES block length plus 1 for the padding-content indicator byte, i.e. from 17 bytes up to and including 193 bytes. *Note:* See ISO/IEC 7816-4:2013 Table 49 for the SM data object with tag ‘87h’.
		‘81h’ + ‘10h’	DER-TLV encoded Control Reference Template for Confidentiality nesting the concatenation of the following data elements (see Appendix 1 DSRCSecurityData and Appendix 11 Part B chapter 13): — 4 byte time stamp — 3 byte counter — 8 byte VU serial number — 1 byte DSRC master key version *Note:* See ISO/IEC 7816-4:2013 Table 49 for the SM data object with tag ‘81h’.
		‘8Eh’ + L_{8E} + ‘XX..XXh’	DER-TLV encoded MAC over the DSRC message. For the

			MAC algorithm and calculation see Appendix 11, Part B chapter 13. *Note:* See ISO/IEC 7816-4:2013 Table 49 for the SM data object with tag '8Eh'.
[F15 + L + 1	1	'00h'	As specified in ISO/IEC 7816-4]

TCS_139**Response Message**

Byte	Length	Value	Description
#1-#L	L	'XX..XXh'	Absent (in case of an error) or deciphered data (padding removed)
SW	2	'XXXXh'	Status Words (SW1,SW2)

- If the command is successful, the card returns '**9000**'.
- '**6A80**' indicates incorrect parameters in the command data field (also used in case the data objects are not sent in the specified order).
- '**6A88**' indicates that referenced data is not available, i.e. the referenced DSRC master key is not available.
- '**6900**' indicates that the verification of the cryptographic checksum or the decryption of the data failed.
- '[F1**6985**' indicates that the 4-byte time stamp provided in the command data field is earlier than cardValidityBegin or later than cardExpiryDate.]

4. TACHOGRAPH CARDS STRUCTURE

This paragraph specifies the file structures of the Tachograph cards for storage of accessible data.

It does not specify card manufacturer dependent internal structures, such as e.g. file headers, nor storage and handling of data elements needed for internal use only such as `EuropeanPublicKey`,`CardPrivateKey`,`TdesSessionKey` or`WorkshopCardPin`.

TCS_140A generation 2 tachograph card shall host the Master File MF and a generation 1 and a generation 2 tachograph application of the same type (e.g. driver card applications).

TCS_141A tachograph card shall support at least the minimum number of records specified for the corresponding applications and shall not support more records than the maximum number of records specified for the corresponding applications.

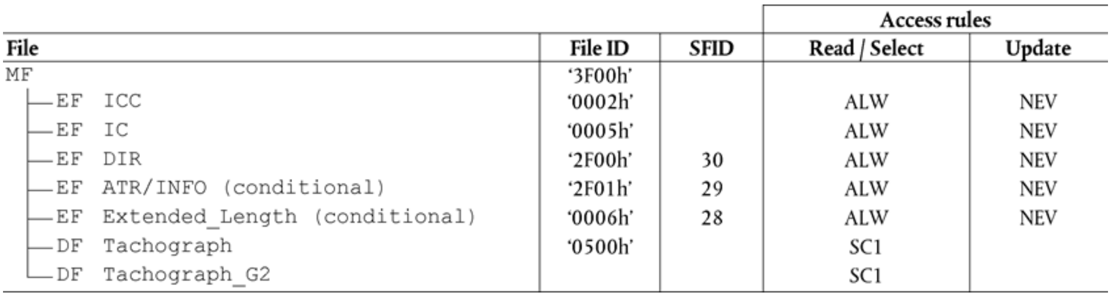
The maximum and minimum numbers of records are specified in this chapter for the different applications.

For the security conditions used in the access rules throughout this chapter please refer to chapter 3.3. In general the access mode ‘read’ denotes the READ BINARY command with even and if supported odd INS byte with the exception of the EF Sensor_Installation_Data on the workshop card, see TCS_156 and TCS_160. The access mode ‘update’ denotes the Update Binary command with even and if supported odd INS byte and the access mode ‘select’ the SELECT command.

4.1. Master File MF

TCS_142After its personalisation, the master file MF shall have the following permanent file structure and file access rules:

Note: The short EF identifier SFID is given as decimal number, e.g. the value 30 corresponds to 11110 in binary.



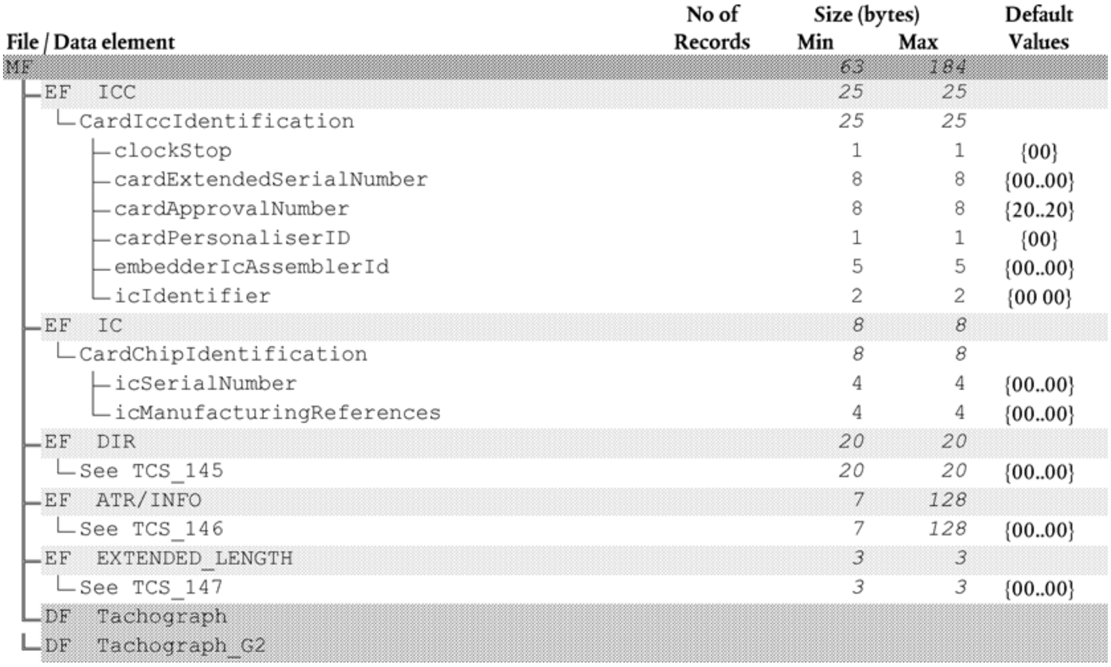
			Access rules	
File	File ID	SFID	Read / Select	Update
MF	‘3F00h’			
├─EF ICC	‘0002h’		ALW	NEV
├─EF IC	‘0005h’		ALW	NEV
├─EF DIR	‘2F00h’	30	ALW	NEV
├─EF ATR/INFO (conditional)	‘2F01h’	29	ALW	NEV
├─EF Extended_Length (conditional)	‘0006h’	28	ALW	NEV
├─DF Tachograph	‘0500h’		SC1	
└─DF Tachograph_G2			SC1	

The following abbreviation for the security condition is used in this table:

SC1 ALW OR SM-MAC-G2

TCS_143All EF structures shall be transparent.

TCS_144The Master File MF shall have the following data structure:



	No of	Size (bytes)		Default
File / Data element	Records	Min	Max	Values
MF		63	184	
├─EF ICC		25	25	
│ └─CardIccIdentification		25	25	
│ ├─clockStop		1	1	{00}
│ ├─cardExtendedSerialNumber		8	8	{00..00}
│ ├─cardApprovalNumber		8	8	{20..20}
│ ├─cardPersonaliserID		1	1	{00}
│ ├─embedderIcAssemblerId		5	5	{00..00}
│ └─icIdentifier		2	2	{00 00}
├─EF IC		8	8	
│ └─CardChipIdentification		8	8	
│ ├─icSerialNumber		4	4	{00..00}
│ └─icManufacturingReferences		4	4	{00..00}
├─EF DIR		20	20	
│ └─See TCS_145		20	20	{00..00}
├─EF ATR/INFO		7	128	
│ └─See TCS_146		7	128	{00..00}
├─EF EXTENDED_LENGTH		3	3	
│ └─See TCS_147		3	3	{00..00}
├─DF Tachograph				
└─DF Tachograph_G2				

TCS_145The elementary file EF DIR shall contain the following application related data objects: '61 08 4F 06 FF 54 41 43 48 4F 61 08 4F 06 FF 53 4D 52 44 54'

TCS_146The elementary file EF ATR/INFO shall be present if the tachograph card indicates in its ATR that it supports extended length fields. In this case the EF ATR/INFO shall contain the extended length information data object (DO'7F66') as specified in ISO/ IEC 7816-4:2013 clause 12.7.1.

TCS_147The elementary file EF Extended_Length shall be present if the tachograph card indicates in its ATR that it supports extended length fields. In this case the EF shall contain the following data object: '02 01 xx' where the value 'xx' indicates whether extended length fields are supported for the T = 1 and / or T = 0 protocol.

The value '01' indicates extended length field support for the T = 1 protocol.

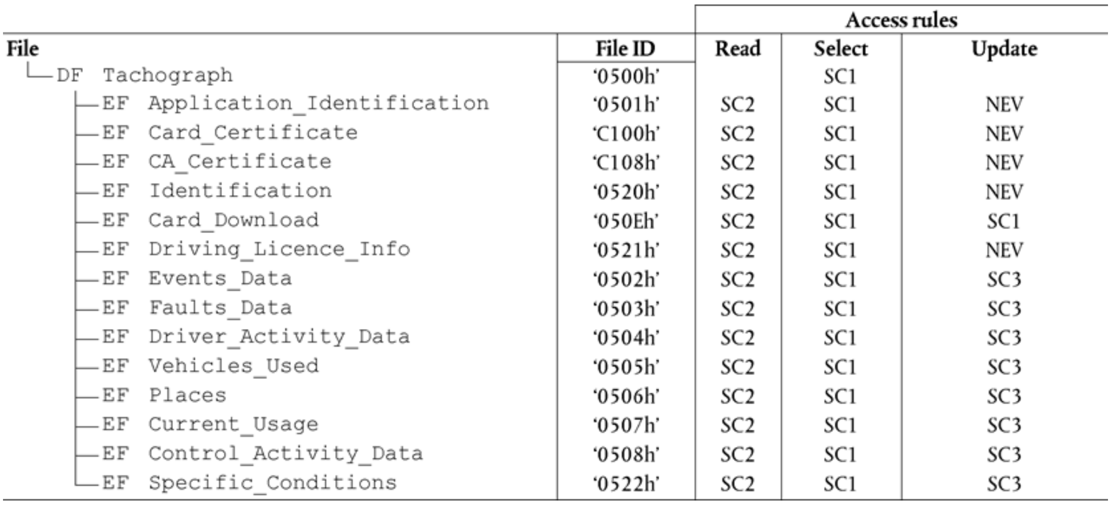
The value '10' indicates extended length field support for the T = 0 protocol.

The value '11' indicates extended length field support for the T = 1 and the T = 0 protocol.

4.2. **Driver card applications**

4.2.1 *Driver card application generation 1*

TCS_148After its personalisation, the driver card application generation 1 shall have the following permanent file structure and file access rules:



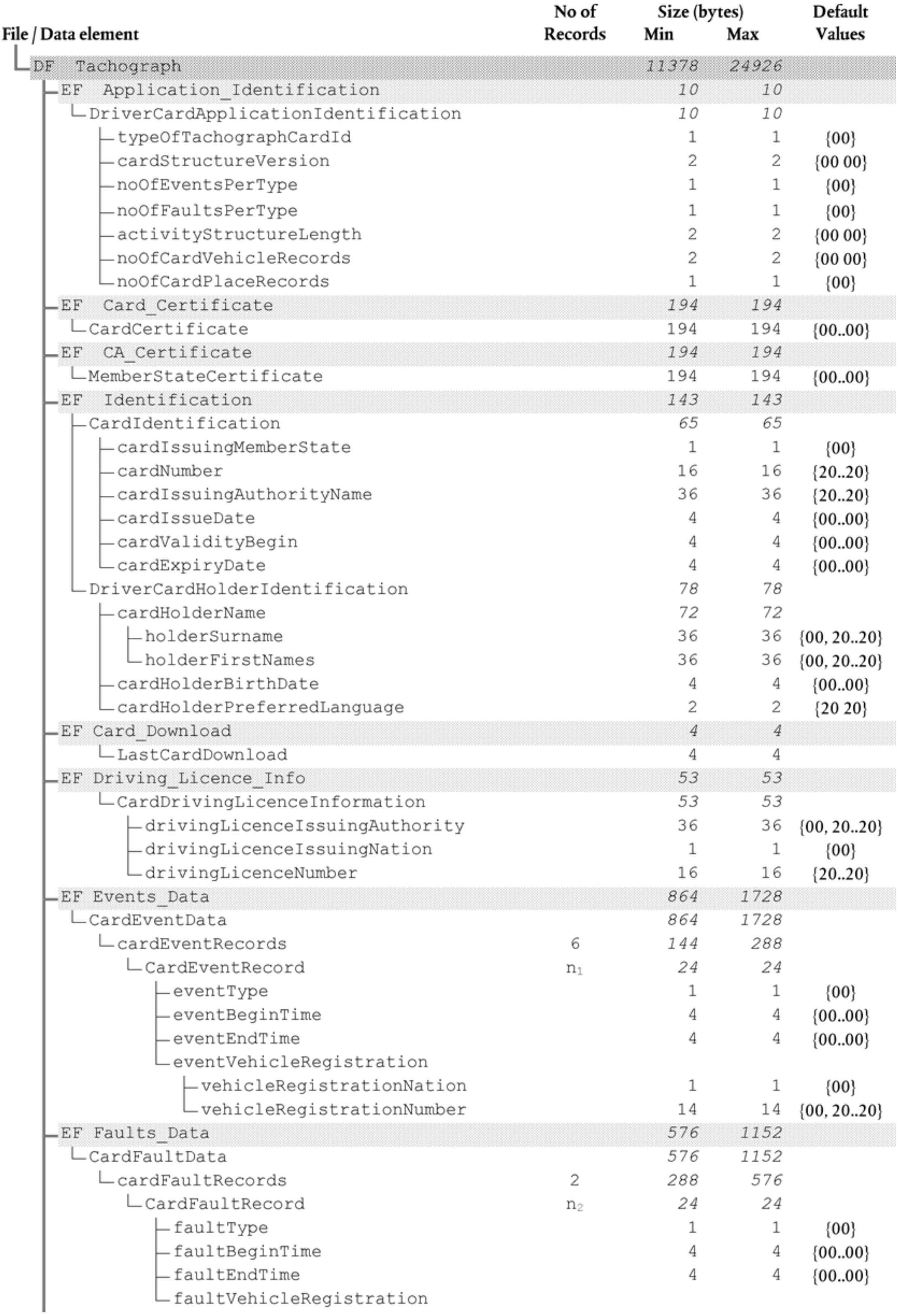
		Access rules		
File	**File ID**	**Read**	**Select**	**Update**
└─DF Tachograph	'0500h'		SC1	
├─EF Application_Identification	'0501h'	SC2	SC1	NEV
├─EF Card_Certificate	'C100h'	SC2	SC1	NEV
├─EF CA_Certificate	'C108h'	SC2	SC1	NEV
├─EF Identification	'0520h'	SC2	SC1	NEV
├─EF Card_Download	'050Eh'	SC2	SC1	SC1
├─EF Driving_Licence_Info	'0521h'	SC2	SC1	NEV
├─EF Events_Data	'0502h'	SC2	SC1	SC3
├─EF Faults_Data	'0503h'	SC2	SC1	SC3
├─EF Driver_Activity_Data	'0504h'	SC2	SC1	SC3
├─EF Vehicles_Used	'0505h'	SC2	SC1	SC3
├─EF Places	'0506h'	SC2	SC1	SC3
├─EF Current_Usage	'0507h'	SC2	SC1	SC3
├─EF Control_Activity_Data	'0508h'	SC2	SC1	SC3
└─EF Specific_Conditions	'0522h'	SC2	SC1	SC3

The following abbreviations for the security conditions are used in this table:

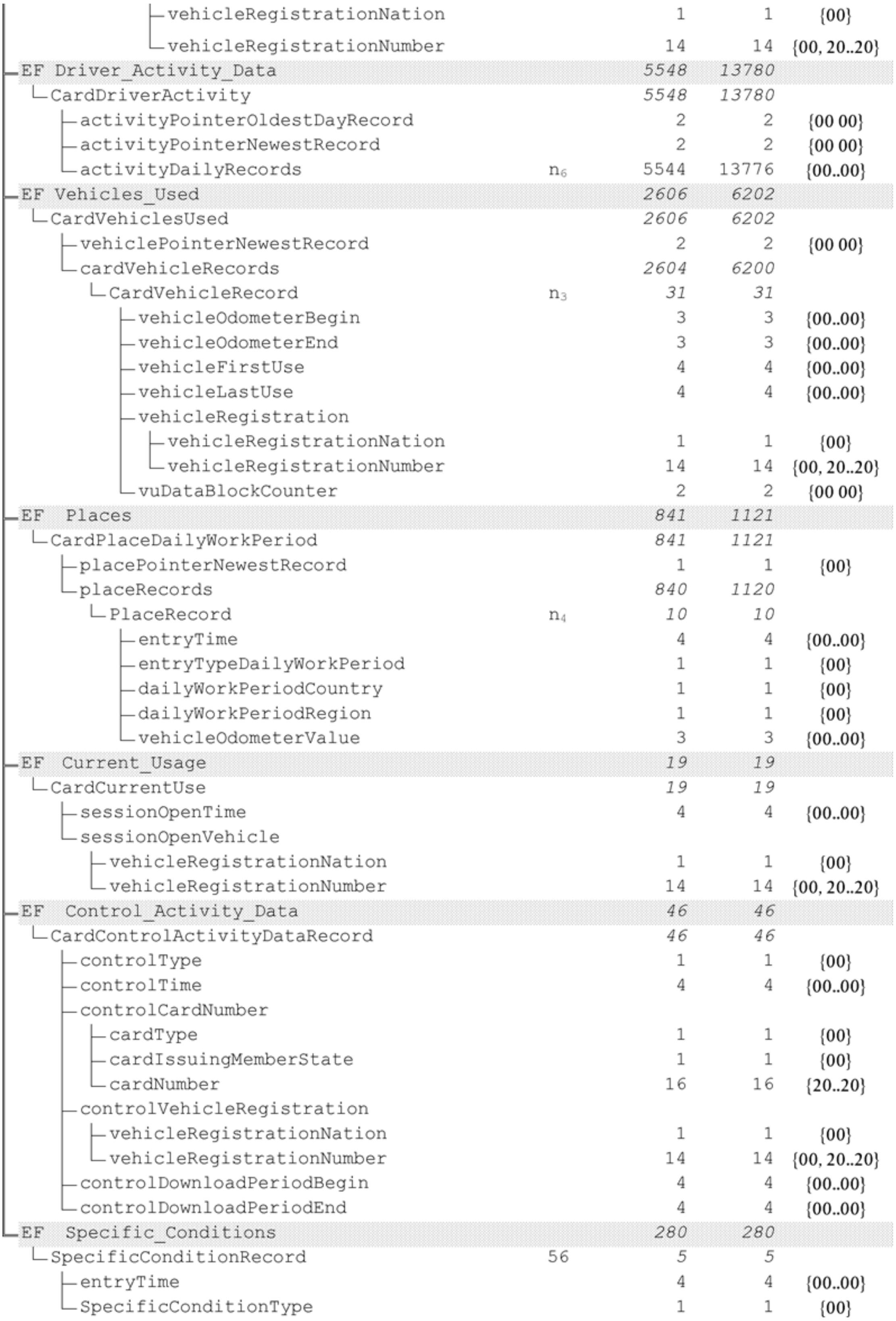
SC1	ALW OR SM-MAC-G2
SC2	ALW OR SM-MAC-G1 OR SM-MAC-G2
SC3	SM-MAC-G1 OR SM-MAC-G2

TCS_149All EF structures shall be transparent.

TCS_150The driver card application generation 1 shall have the following data structure:



File / Data element	No of Records	Size (bytes)		Default Values
		Min	Max	
└DF Tachograph		*11378*	*24926*	
├EF Application_Identification		*10*	*10*	
│ └DriverCardApplicationIdentification		*10*	*10*	
│ │ ├typeOfTachographCardId		1	1	{00}
│ │ ├cardStructureVersion		2	2	{00 00}
│ │ ├noOfEventsPerType		1	1	{00}
│ │ ├noOfFaultsPerType		1	1	{00}
│ │ ├activityStructureLength		2	2	{00 00}
│ │ ├noOfCardVehicleRecords		2	2	{00 00}
│ │ └noOfCardPlaceRecords		1	1	{00}
├EF Card_Certificate		*194*	*194*	
│ └CardCertificate		194	194	{00..00}
├EF CA_Certificate		*194*	*194*	
│ └MemberStateCertificate		194	194	{00..00}
├EF Identification		*143*	*143*	
│ ├CardIdentification		*65*	*65*	
│ │ ├cardIssuingMemberState		1	1	{00}
│ │ ├cardNumber		16	16	{20..20}
│ │ ├cardIssuingAuthorityName		36	36	{20..20}
│ │ ├cardIssueDate		4	4	{00..00}
│ │ ├cardValidityBegin		4	4	{00..00}
│ │ └cardExpiryDate		4	4	{00..00}
│ └DriverCardHolderIdentification		*78*	*78*	
│ │ ├cardHolderName		*72*	*72*	
│ │ │ ├holderSurname		36	36	{00, 20..20}
│ │ │ └holderFirstNames		36	36	{00, 20..20}
│ │ ├cardHolderBirthDate		4	4	{00..00}
│ │ └cardHolderPreferredLanguage		2	2	{20 20}
├EF Card_Download		*4*	*4*	
│ │ └LastCardDownload		4	4	
├EF Driving_Licence_Info		*53*	*53*	
│ │ └CardDrivingLicenceInformation		*53*	*53*	
│ │ │ ├drivingLicenceIssuingAuthority		36	36	{00, 20..20}
│ │ │ ├drivingLicenceIssuingNation		1	1	{00}
│ │ │ └drivingLicenceNumber		16	16	{20..20}
├EF Events_Data		*864*	*1728*	
│ └CardEventData		*864*	*1728*	
│ │ └cardEventRecords	6	*144*	*288*	
│ │ │ └CardEventRecord	n_1	*24*	*24*	
│ │ │ │ ├eventType		1	1	{00}
│ │ │ │ ├eventBeginTime		4	4	{00..00}
│ │ │ │ ├eventEndTime		4	4	{00..00}
│ │ │ │ └eventVehicleRegistration				
│ │ │ │ │ ├vehicleRegistrationNation		1	1	{00}
│ │ │ │ │ └vehicleRegistrationNumber		14	14	{00, 20..20}
├EF Faults_Data		*576*	*1152*	
│ └CardFaultData		*576*	*1152*	
│ │ └cardFaultRecords	2	*288*	*576*	
│ │ │ └CardFaultRecord	n_2	*24*	*24*	
│ │ │ │ ├faultType		1	1	{00}
│ │ │ │ ├faultBeginTime		4	4	{00..00}
│ │ │ │ ├faultEndTime		4	4	{00..00}
│ │ │ │ └faultVehicleRegistration				



├─vehicleRegistrationNation		1	1	{00}
└─vehicleRegistrationNumber		14	14	{00, 20..20}
─EF Driver_Activity_Data		*5548*	*13780*	
└─CardDriverActivity		*5548*	*13780*	
├─activityPointerOldestDayRecord		2	2	{00 00}
├─activityPointerNewestRecord		2	2	{00 00}
└─activityDailyRecords	n_6	5544	13776	{00..00}
─EF Vehicles_Used		*2606*	*6202*	
└─CardVehiclesUsed		*2606*	*6202*	
├─vehiclePointerNewestRecord		2	2	{00 00}
└─cardVehicleRecords		*2604*	*6200*	
└─CardVehicleRecord	n_3	*31*	*31*	
├─vehicleOdometerBegin		3	3	{00..00}
├─vehicleOdometerEnd		3	3	{00..00}
├─vehicleFirstUse		4	4	{00..00}
├─vehicleLastUse		4	4	{00..00}
├─vehicleRegistration				
├─vehicleRegistrationNation		1	1	{00}
└─vehicleRegistrationNumber		14	14	{00, 20..20}
└─vuDataBlockCounter		2	2	{00 00}
─EF Places		*841*	*1121*	
└─CardPlaceDailyWorkPeriod		*841*	*1121*	
├─placePointerNewestRecord		1	1	{00}
└─placeRecords		*840*	*1120*	
└─PlaceRecord	n_4	*10*	*10*	
├─entryTime		4	4	{00..00}
├─entryTypeDailyWorkPeriod		1	1	{00}
├─dailyWorkPeriodCountry		1	1	{00}
├─dailyWorkPeriodRegion		1	1	{00}
└─vehicleOdometerValue		3	3	{00..00}
─EF Current_Usage		*19*	*19*	
└─CardCurrentUse		*19*	*19*	
├─sessionOpenTime		4	4	{00..00}
└─sessionOpenVehicle				
├─vehicleRegistrationNation		1	1	{00}
└─vehicleRegistrationNumber		14	14	{00, 20..20}
─EF Control_Activity_Data		*46*	*46*	
└─CardControlActivityDataRecord		*46*	*46*	
├─controlType		1	1	{00}
├─controlTime		4	4	{00..00}
├─controlCardNumber				
├─cardType		1	1	{00}
├─cardIssuingMemberState		1	1	{00}
└─cardNumber		16	16	{20..20}
├─controlVehicleRegistration				
├─vehicleRegistrationNation		1	1	{00}
└─vehicleRegistrationNumber		14	14	{00, 20..20}
├─controlDownloadPeriodBegin		4	4	{00..00}
└─controlDownloadPeriodEnd		4	4	{00..00}
─EF Specific_Conditions		*280*	*280*	
└─SpecificConditionRecord	56	*5*	*5*	
├─entryTime		4	4	{00..00}
└─SpecificConditionType		1	1	{00}

TCS_151The following values, used to provide sizes in the table above, are the minimum and maximum record number values the driver card data structure must use for a generation 1 application:

		Min	Max
n_1	NoOfEventsPerType	6	12
n_2	NoOfFaultsPerType	12	24
n_3	NoOfCardVehicleRecords	84	200
n_4	NoOfCardPlaceRecords	84	112
n_6	CardActivityLengthRange	5 544 bytes (28 days * 93 activity changes)	13 776 Bytes (28 days * 240 activity changes)

4.2.2 *Driver card application generation 2*

TCS_152After its personalisation, the driver card application generation 2 shall have the following permanent file structure and file access rules.

Note: The short EF identifier SFID is given as decimal number, e.g. the value 30 corresponds to 11110 in binary.

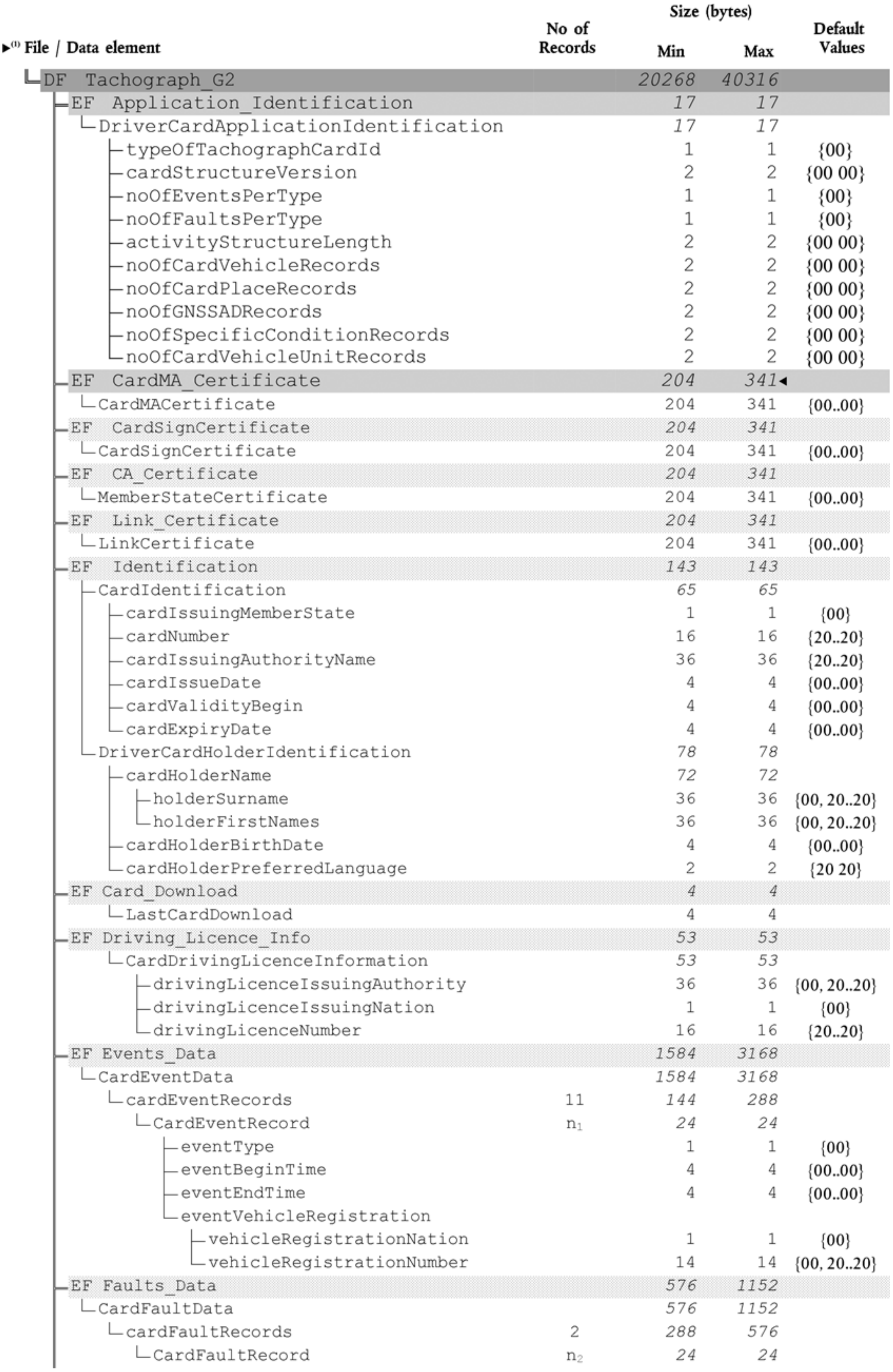
			Access rules	
File	**File ID**	**SFID**	**Read / Select**	**Update**
└─DF Tachograph_G2			SC1	
├─EF Application_Identification	'0501h'	1	SC1	NEV
├─EF CardMA_Certificate	'C100h'	2	SC1	NEV
├─EF CardSignCertificate	'C101h'	3	SC1	NEV
├─EF CA_Certificate	'C108h'	4	SC1	NEV
├─EF Link_Certificate	'C109h'	5	SC1	NEV
├─EF Identification	'0520h'	6	SC1	NEV
├─EF Card_Download	'050Eh'	7	SC1	SC1
├─EF Driving_Licence_Info	'0521h'	10	SC1	NEV
├─EF Events_Data	'0502h'	12	SC1	SM-MAC-G2
├─EF Faults_Data	'0503h'	13	SC1	SM-MAC-G2
├─EF Driver_Activity_Data	'0504h'	14	SC1	SM-MAC-G2
├─EF Vehicles_Used	'0505h'	15	SC1	SM-MAC-G2
├─EF Places	'0506h'	16	SC1	SM-MAC-G2
├─EF Current_Usage	'0507h'	17	SC1	SM-MAC-G2
├─EF Control_Activity_Data	'0508h'	18	SC1	SM-MAC-G2
├─EF Specific_Conditions	'0522h'	19	SC1	SM-MAC-G2
├─EF VehicleUnits_Used	'0523h'	20	SC1	SM-MAC-G2
└─EF GNSS_Places	'0524h'	21	SC1	SM-MAC-G2

The following abbreviation for the security condition is used in this table:

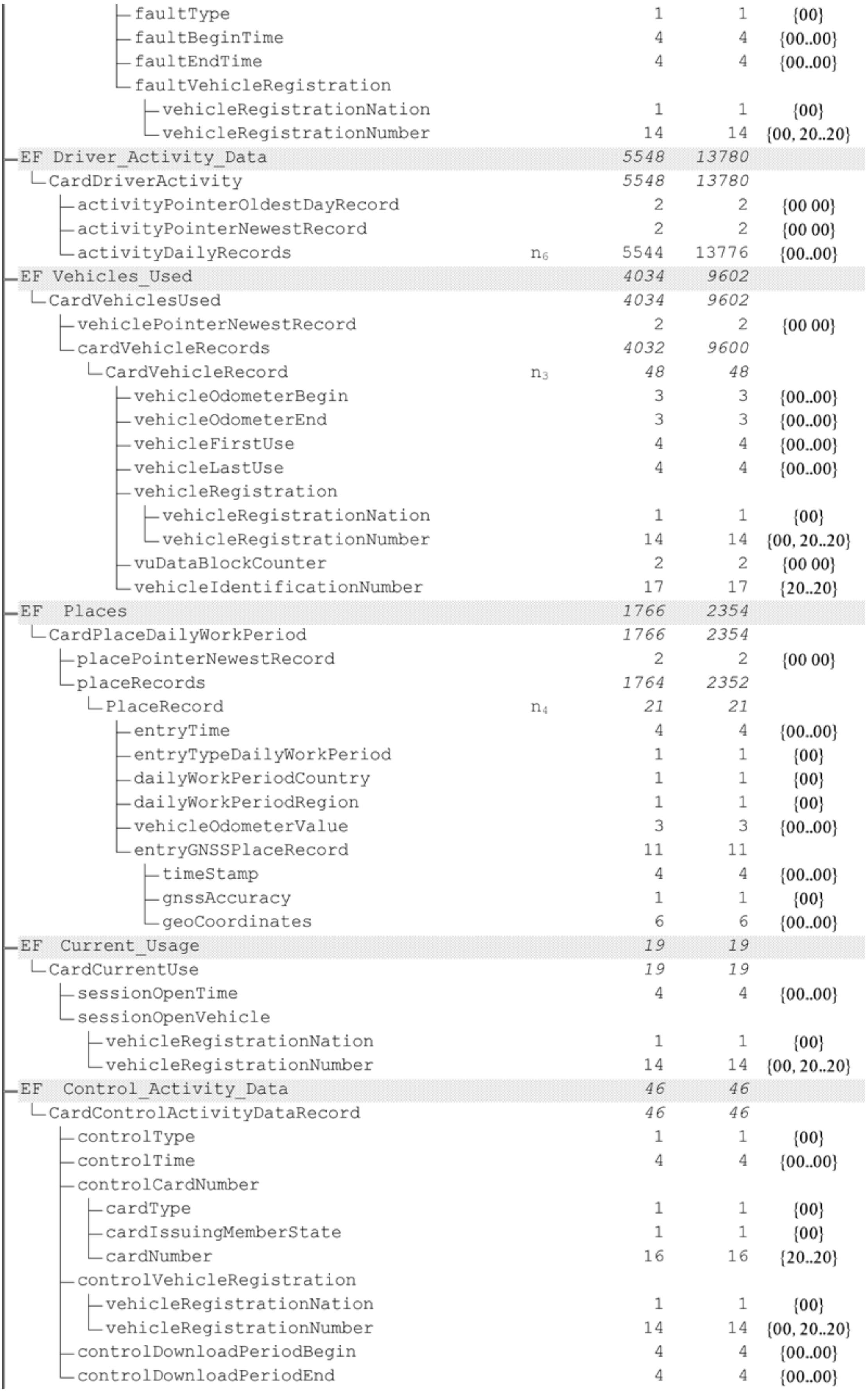
SC1	ALW OR SM-MAC-G2

TCS_153All EF structures shall be transparent.

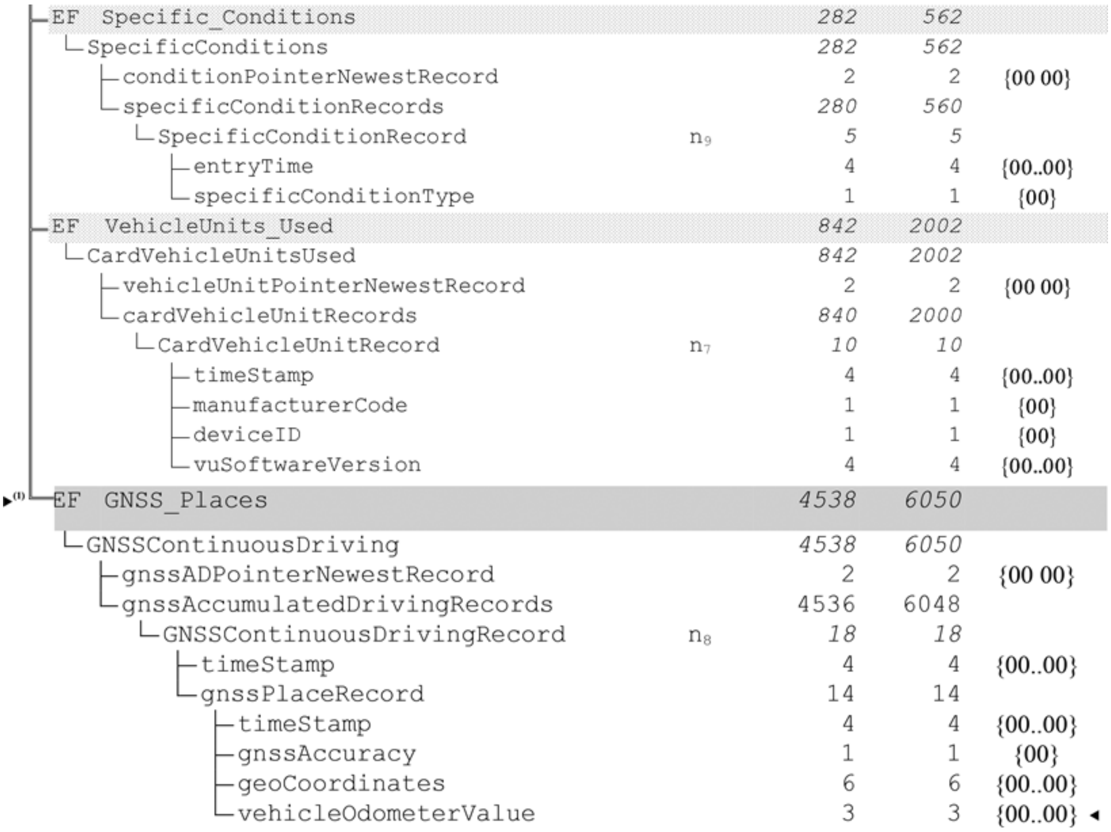
TCS_154The driver card application generation 2 shall have the following data structure:



▸[1] File / Data element	No of Records	Size (bytes)		Default Values
		Min	Max	
└DF Tachograph_G2		20268	40316	
─EF Application_Identification		17	17	
└DriverCardApplicationIdentification		17	17	
├typeOfTachographCardId		1	1	{00}
├cardStructureVersion		2	2	{00 00}
├noOfEventsPerType		1	1	{00}
├noOfFaultsPerType		1	1	{00}
├activityStructureLength		2	2	{00 00}
├noOfCardVehicleRecords		2	2	{00 00}
├noOfCardPlaceRecords		2	2	{00 00}
├noOfGNSSADRecords		2	2	{00 00}
├noOfSpecificConditionRecords		2	2	{00 00}
└noOfCardVehicleUnitRecords		2	2	{00 00}
─EF CardMA_Certificate		204	341◂	
└CardMACertificate		204	341	{00..00}
─EF CardSignCertificate		204	341	
└CardSignCertificate		204	341	{00..00}
─EF CA_Certificate		204	341	
└MemberStateCertificate		204	341	{00..00}
─EF Link_Certificate		204	341	
└LinkCertificate		204	341	{00..00}
─EF Identification		143	143	
├CardIdentification		65	65	
├cardIssuingMemberState		1	1	{00}
├cardNumber		16	16	{20..20}
├cardIssuingAuthorityName		36	36	{20..20}
├cardIssueDate		4	4	{00..00}
├cardValidityBegin		4	4	{00..00}
└cardExpiryDate		4	4	{00..00}
└DriverCardHolderIdentification		78	78	
├cardHolderName		72	72	
├holderSurname		36	36	{00, 20..20}
└holderFirstNames		36	36	{00, 20..20}
├cardHolderBirthDate		4	4	{00..00}
└cardHolderPreferredLanguage		2	2	{20 20}
─EF Card_Download		4	4	
└LastCardDownload		4	4	
─EF Driving_Licence_Info		53	53	
└CardDrivingLicenceInformation		53	53	
├drivingLicenceIssuingAuthority		36	36	{00, 20..20}
├drivingLicenceIssuingNation		1	1	{00}
└drivingLicenceNumber		16	16	{20..20}
─EF Events_Data		1584	3168	
└CardEventData		1584	3168	
└cardEventRecords	11	144	288	
└CardEventRecord	n_1	24	24	
├eventType		1	1	{00}
├eventBeginTime		4	4	{00..00}
├eventEndTime		4	4	{00..00}
└eventVehicleRegistration				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
─EF Faults_Data		576	1152	
└CardFaultData		576	1152	
└cardFaultRecords	2	288	576	
└CardFaultRecord	n_2	24	24	

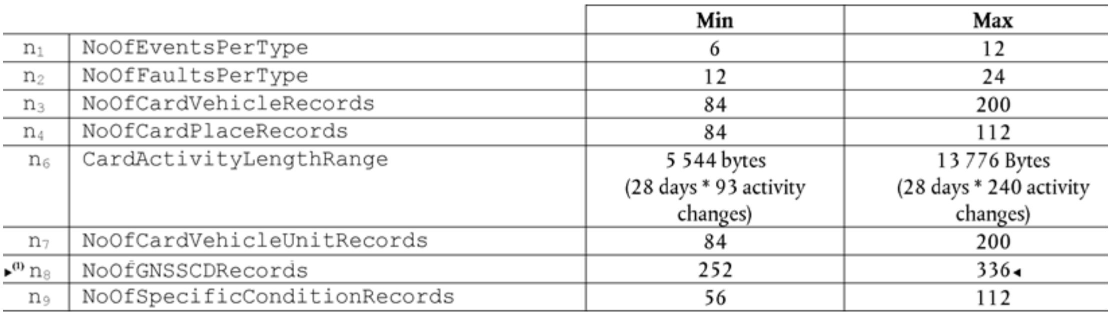


```
│         ├─faultType                                       1      1   {00}
│         ├─faultBeginTime                                  4      4   {00..00}
│         ├─faultEndTime                                    4      4   {00..00}
│         └─faultVehicleRegistration
│            ├─vehicleRegistrationNation                    1      1   {00}
│            └─vehicleRegistrationNumber                   14     14   {00, 20..20}
├─EF Driver_Activity_Data                                5548  13780
│  └─CardDriverActivity                                  5548  13780
│     ├─activityPointerOldestDayRecord                      2      2   {00 00}
│     ├─activityPointerNewestRecord                         2      2   {00 00}
│     └─activityDailyRecords                     n6      5544  13776   {00..00}
├─EF Vehicles_Used                                       4034   9602
│  └─CardVehiclesUsed                                    4034   9602
│     ├─vehiclePointerNewestRecord                          2      2   {00 00}
│     └─cardVehicleRecords                               4032   9600
│        └─CardVehicleRecord                     n3        48     48
│           ├─vehicleOdometerBegin                          3      3   {00..00}
│           ├─vehicleOdometerEnd                            3      3   {00..00}
│           ├─vehicleFirstUse                               4      4   {00..00}
│           ├─vehicleLastUse                                4      4   {00..00}
│           ├─vehicleRegistration
│           │  ├─vehicleRegistrationNation                  1      1   {00}
│           │  └─vehicleRegistrationNumber                 14     14   {00, 20..20}
│           ├─vuDataBlockCounter                            2      2   {00 00}
│           └─vehicleIdentificationNumber                  17     17   {20..20}
├─EF  Places                                             1766   2354
│  └─CardPlaceDailyWorkPeriod                            1766   2354
│     ├─placePointerNewestRecord                            2      2   {00 00}
│     └─placeRecords                                     1764   2352
│        └─PlaceRecord                           n4        21     21
│           ├─entryTime                                     4      4   {00..00}
│           ├─entryTypeDailyWorkPeriod                      1      1   {00}
│           ├─dailyWorkPeriodCountry                        1      1   {00}
│           ├─dailyWorkPeriodRegion                         1      1   {00}
│           ├─vehicleOdometerValue                          3      3   {00..00}
│           └─entryGNSSPlaceRecord                         11     11
│              ├─timeStamp                                  4      4   {00..00}
│              ├─gnssAccuracy                               1      1   {00}
│              └─geoCoordinates                             6      6   {00..00}
├─EF  Current_Usage                                        19     19
│  └─CardCurrentUse                                        19     19
│     ├─sessionOpenTime                                     4      4   {00..00}
│     └─sessionOpenVehicle
│        ├─vehicleRegistrationNation                        1      1   {00}
│        └─vehicleRegistrationNumber                       14     14   {00, 20..20}
├─EF  Control_Activity_Data                                46     46
│  └─CardControlActivityDataRecord                         46     46
│     ├─controlType                                         1      1   {00}
│     ├─controlTime                                         4      4   {00..00}
│     ├─controlCardNumber
│     │  ├─cardType                                         1      1   {00}
│     │  ├─cardIssuingMemberState                           1      1   {00}
│     │  └─cardNumber                                      16     16   {20..20}
│     ├─controlVehicleRegistration
│     │  ├─vehicleRegistrationNation                        1      1   {00}
│     │  └─vehicleRegistrationNumber                       14     14   {00, 20..20}
│     ├─controlDownloadPeriodBegin                          4      4   {00..00}
│     └─controlDownloadPeriodEnd                            4      4   {00..00}
```



```
 ├EF  Specific_Conditions                                 282    562
 │ └SpecificConditions                                    282    562
 │   ├conditionPointerNewestRecord                          2      2   {00 00}
 │   └specificConditionRecords                            280    560
 │     └SpecificConditionRecord                  n9         5      5
 │       ├entryTime                                         4      4   {00..00}
 │       └specificConditionType                             1      1   {00}
 ├EF  VehicleUnits_Used                                   842   2002
 │ └CardVehicleUnitsUsed                                  842   2002
 │   ├vehicleUnitPointerNewestRecord                        2      2   {00 00}
 │   └cardVehicleUnitRecords                              840   2000
 │     └CardVehicleUnitRecord                    n7        10     10
 │       ├timeStamp                                         4      4   {00..00}
 │       ├manufacturerCode                                  1      1   {00}
 │       ├deviceID                                          1      1   {00}
 │       └vuSoftwareVersion                                 4      4   {00..00}
▸(1)└EF  GNSS_Places                                     4538   6050
   └GNSSContinuousDriving                                4538   6050
     ├gnssADPointerNewestRecord                             2      2   {00 00}
     └gnssAccumulatedDrivingRecords                      4536   6048
       └GNSSContinuousDrivingRecord              n8        18     18
         ├timeStamp                                         4      4   {00..00}
         └gnssPlaceRecord                                  14     14
           ├timeStamp                                       4      4   {00..00}
           ├gnssAccuracy                                    1      1   {00}
           ├geoCoordinates                                  6      6   {00..00}
           └vehicleOdometerValue                            3      3   {00..00} ◂
```

TCS_155 The following values, used to provide sizes in the table above, are the minimum and maximum record number values the driver card data structure must use for a generation 2 application:

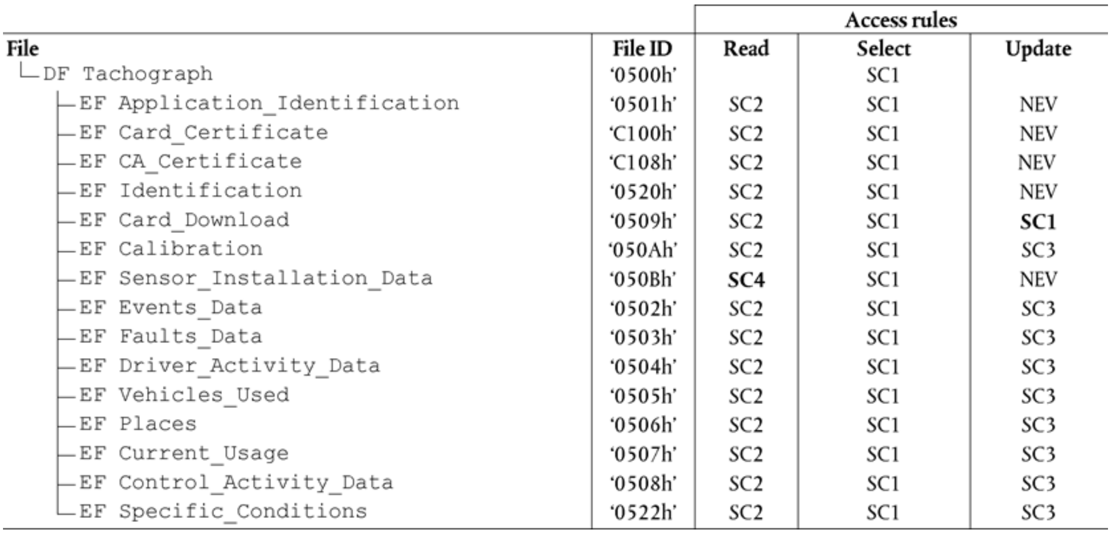


		Min	Max
n_1	NoOfEventsPerType	6	12
n_2	NoOfFaultsPerType	12	24
n_3	NoOfCardVehicleRecords	84	200
n_4	NoOfCardPlaceRecords	84	112
n_6	CardActivityLengthRange	5 544 bytes (28 days * 93 activity changes)	13 776 Bytes (28 days * 240 activity changes)
n_7	NoOfCardVehicleUnitRecords	84	200
▸(1) n_8	NoOfGNSSCDRecords	252	336◂
n_9	NoOfSpecificConditionRecords	56	112

4.3. Workshop card applications

4.3.1 *Workshop card application generation 1*

TCS_156 After its personalisation, the workshop card application generation 1 shall have the following permanent file structure and file access rules:



		Access rules		
File	**File ID**	**Read**	**Select**	**Update**
└─DF Tachograph	'0500h'		SC1	
├─EF Application_Identification	'0501h'	SC2	SC1	NEV
├─EF Card_Certificate	'C100h'	SC2	SC1	NEV
├─EF CA_Certificate	'C108h'	SC2	SC1	NEV
├─EF Identification	'0520h'	SC2	SC1	NEV
├─EF Card_Download	'0509h'	SC2	SC1	**SC1**
├─EF Calibration	'050Ah'	SC2	SC1	SC3
├─EF Sensor_Installation_Data	'050Bh'	**SC4**	SC1	NEV
├─EF Events_Data	'0502h'	SC2	SC1	SC3
├─EF Faults_Data	'0503h'	SC2	SC1	SC3
├─EF Driver_Activity_Data	'0504h'	SC2	SC1	SC3
├─EF Vehicles_Used	'0505h'	SC2	SC1	SC3
├─EF Places	'0506h'	SC2	SC1	SC3
├─EF Current_Usage	'0507h'	SC2	SC1	SC3
├─EF Control_Activity_Data	'0508h'	SC2	SC1	SC3
└─EF Specific_Conditions	'0522h'	SC2	SC1	SC3

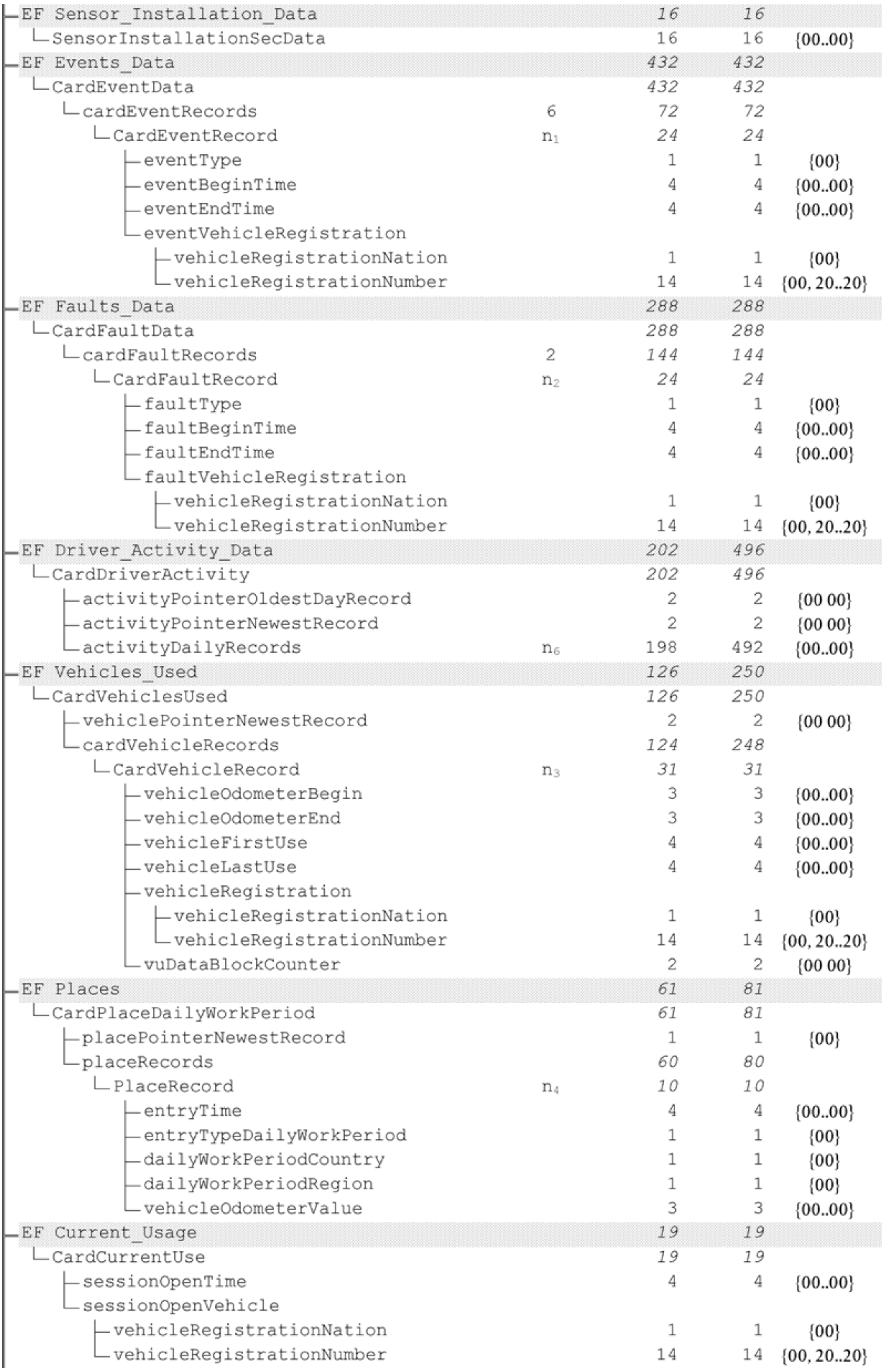
The following abbreviations for the security conditions are used in this table:

SC1	ALW OR SM-MAC-G2
SC2	ALW OR SM-MAC-G1 OR SM-MAC-G2
SC3	SM-MAC-G1 OR SM-MAC-G2
[F2**SC4**	For the READ BINARY command with even INS byte: (SM-C-MAC-G1 AND SM-R-ENC-MAC-G1) OR (SM-C-MAC-G2 AND SM-R-ENC-MAC-G2) For the READ BINARY command with odd INS byte (if supported): NEV**]**

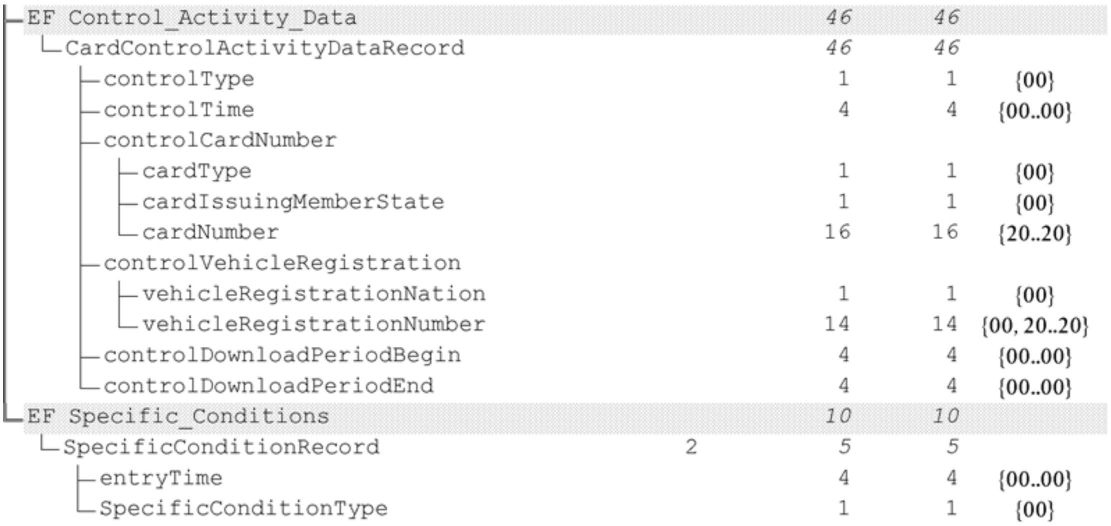
TCS_157All EF structures shall be transparent.

TCS_158The workshop card application generation 1 shall have the following data structure:

File / Data element	No of Records	Size (Bytes)		Default Values
		Min	Max	
└─DF Tachograph		*11055*	*29028*	
├─EF Application_Identification		*11*	*11*	
│ └─WorkshopCardApplicationIdentification		*11*	*11*	
│ ├─typeOfTachographCardId		1	1	{00}
│ ├─cardStructureVersion		2	2	{00 00}
│ ├─noOfEventsPerType		1	1	{00}
│ ├─noOfFaultsPerType		1	1	{00}
│ ├─activityStructureLength		2	2	{00 00}
│ ├─noOfCardVehicleRecords		2	2	{00 00}
│ ├─noOfCardPlaceRecords		1	1	{00}
│ └─noOfCalibrationRecords		1	1	{00}
├─EF Card_Certificate		*194*	*194*	
│ └─CardCertificate		194	194	{00..00}
├─EF CA_Certificate		*194*	*194*	
│ └─MemberStateCertificate		194	194	{00..00}
├─EF Identification		*211*	*211*	
│ ├─CardIdentification		65	65	
│ │ ├─cardIssuingMemberState		1	1	{00}
│ │ ├─cardNumber		16	16	{20..20}
│ │ ├─cardIssuingAuthorityName		36	36	{00, 20..20}
│ │ ├─cardIssueDate		4	4	{00..00}
│ │ ├─cardValidityBegin		4	4	{00..00}
│ │ └─cardExpiryDate		4	4	{00..00}
│ └─WorkshopCardHolderIdentification		*146*	*146*	
│ ├─workshopName		36	36	{00, 20..20}
│ ├─workshopAddress		36	36	{00, 20..20}
│ ├─cardHolderName				
│ │ ├─holderSurname		36	36	{00, 20..20}
│ │ └─holderFirstNames		36	36	{00, 20..20}
│ └─cardHolderPreferredLanguage		2	2	{20 20}
├─EF Card_Download		*2*	*2*	
│ └─NoOfCalibrationsSinceDownload		2	2	{00 00}
├─EF Calibration		*9243*	*26778*	
│ └─WorkshopCardCalibrationData		*9243*	*26778*	
│ ├─calibrationTotalNumber		2	2	{00 00}
│ ├─calibrationPointerNewestRecord		1	1	{00}
│ └─calibrationRecords		*9240*	*26775*	
│ └─WorkshopCardCalibrationRecord	n_5	*105*	*105*	
│ ├─calibrationPurpose		1	1	{00}
│ ├─vehicleIdentificationNumber		17	17	{20..20}
│ ├─vehicleRegistration				
│ │ ├─vehicleRegistrationNation		1	1	{00}
│ │ └─vehicleRegistrationNumber		14	14	{00, 20..20}
│ ├─wVehicleCharacteristicConstant		2	2	{00 00}
│ ├─kConstantOfRecordingEquipment		2	2	{00 00}
│ ├─lTyreCircumference		2	2	{00 00}
│ ├─tyreSize		15	15	{20..20}
│ ├─authorisedSpeed		1	1	{00}
│ ├─oldOdometerValue		3	3	{00..00}
│ ├─newOdometerValue		3	3	{00..00}
│ ├─oldTimeValue		4	4	{00..00}
│ ├─newTimeValue		4	4	{00..00}
│ ├─nextCalibrationDate		4	4	{00..00}
│ ├─vuPartNumber		16	16	{20..20}
│ ├─vuSerialNumber		8	8	{00..00}
│ └─sensorSerialNumber		8	8	{00..00}

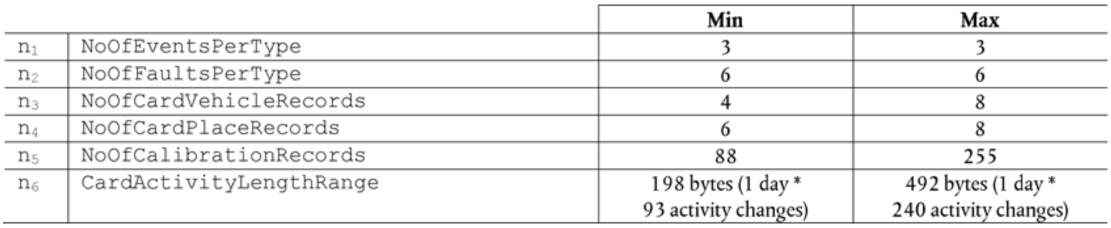


─EF Sensor_Installation_Data		*16*	*16*	
└SensorInstallationSecData		16	16	{00..00}
─EF Events_Data		*432*	*432*	
└CardEventData		*432*	*432*	
└cardEventRecords	6	*72*	*72*	
└CardEventRecord	n_1	*24*	*24*	
├eventType		1	1	{00}
├eventBeginTime		4	4	{00..00}
├eventEndTime		4	4	{00..00}
└eventVehicleRegistration				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
─EF Faults_Data		*288*	*288*	
└CardFaultData		*288*	*288*	
└cardFaultRecords	2	*144*	*144*	
└CardFaultRecord	n_2	*24*	*24*	
├faultType		1	1	{00}
├faultBeginTime		4	4	{00..00}
├faultEndTime		4	4	{00..00}
└faultVehicleRegistration				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
─EF Driver_Activity_Data		*202*	*496*	
└CardDriverActivity		*202*	*496*	
├activityPointerOldestDayRecord		2	2	{00 00}
├activityPointerNewestRecord		2	2	{00 00}
└activityDailyRecords	n_6	198	492	{00..00}
─EF Vehicles_Used		*126*	*250*	
└CardVehiclesUsed		*126*	*250*	
├vehiclePointerNewestRecord		2	2	{00 00}
└cardVehicleRecords		*124*	*248*	
└CardVehicleRecord	n_3	*31*	*31*	
├vehicleOdometerBegin		3	3	{00..00}
├vehicleOdometerEnd		3	3	{00..00}
├vehicleFirstUse		4	4	{00..00}
├vehicleLastUse		4	4	{00..00}
├vehicleRegistration				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
└vuDataBlockCounter		2	2	{00 00}
─EF Places		*61*	*81*	
└CardPlaceDailyWorkPeriod		*61*	*81*	
├placePointerNewestRecord		1	1	{00}
└placeRecords		*60*	*80*	
└PlaceRecord	n_4	*10*	*10*	
├entryTime		4	4	{00..00}
├entryTypeDailyWorkPeriod		1	1	{00}
├dailyWorkPeriodCountry		1	1	{00}
├dailyWorkPeriodRegion		1	1	{00}
└vehicleOdometerValue		3	3	{00..00}
─EF Current_Usage		*19*	*19*	
└CardCurrentUse		*19*	*19*	
├sessionOpenTime		4	4	{00..00}
└sessionOpenVehicle				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}



└EF Control_Activity_Data		*46*	*46*	
└CardControlActivityDataRecord		*46*	*46*	
├controlType		1	1	{00}
├controlTime		4	4	{00..00}
├controlCardNumber				
├cardType		1	1	{00}
├cardIssuingMemberState		1	1	{00}
└cardNumber		16	16	{20..20}
├controlVehicleRegistration				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
├controlDownloadPeriodBegin		4	4	{00..00}
└controlDownloadPeriodEnd		4	4	{00..00}
└EF Specific_Conditions		*10*	*10*	
└SpecificConditionRecord	2	5	5	
├entryTime		4	4	{00..00}
└SpecificConditionType		1	1	{00}

TCS_159The following values, used to provide sizes in the table above, are the minimum and maximum record number values the workshop card data structure must use for a generation 1 application:

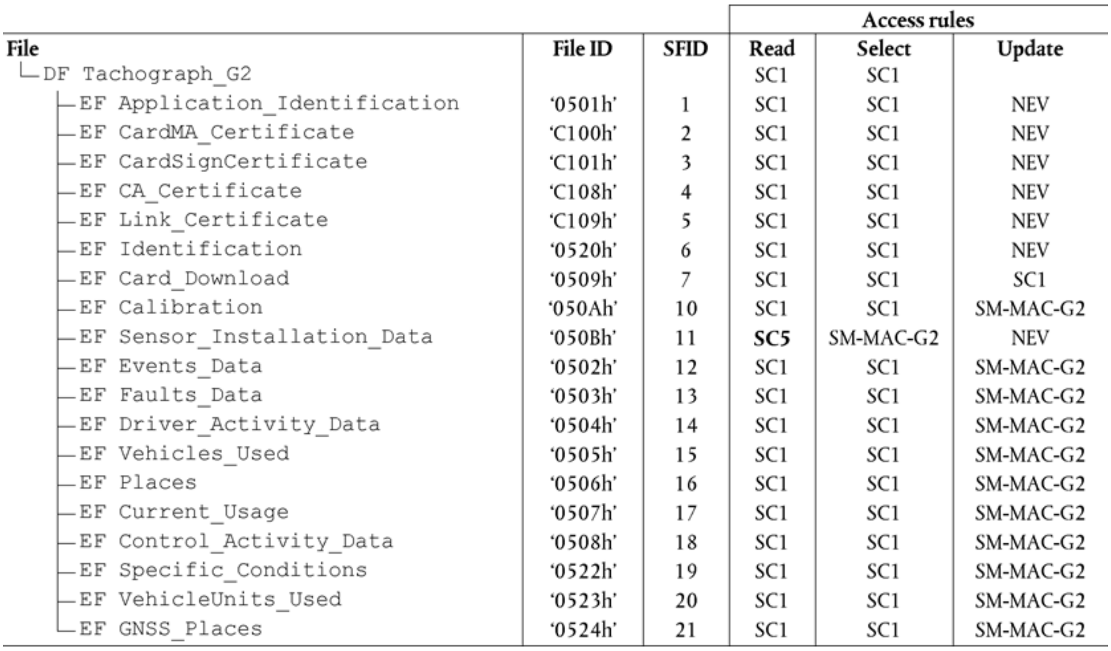


		Min	Max
n_1	NoOfEventsPerType	3	3
n_2	NoOfFaultsPerType	6	6
n_3	NoOfCardVehicleRecords	4	8
n_4	NoOfCardPlaceRecords	6	8
n_5	NoOfCalibrationRecords	88	255
n_6	CardActivityLengthRange	198 bytes (1 day * 93 activity changes)	492 bytes (1 day * 240 activity changes)

4.3.2 *Workshop card application generation 2*

TCS_160After its personalisation, the workshop card application generation 2 shall have the following permanent file structure and file access rules.

Note: The short EF identifier SFID is given as decimal number, e.g. the value 30 corresponds to 11110 in binary.



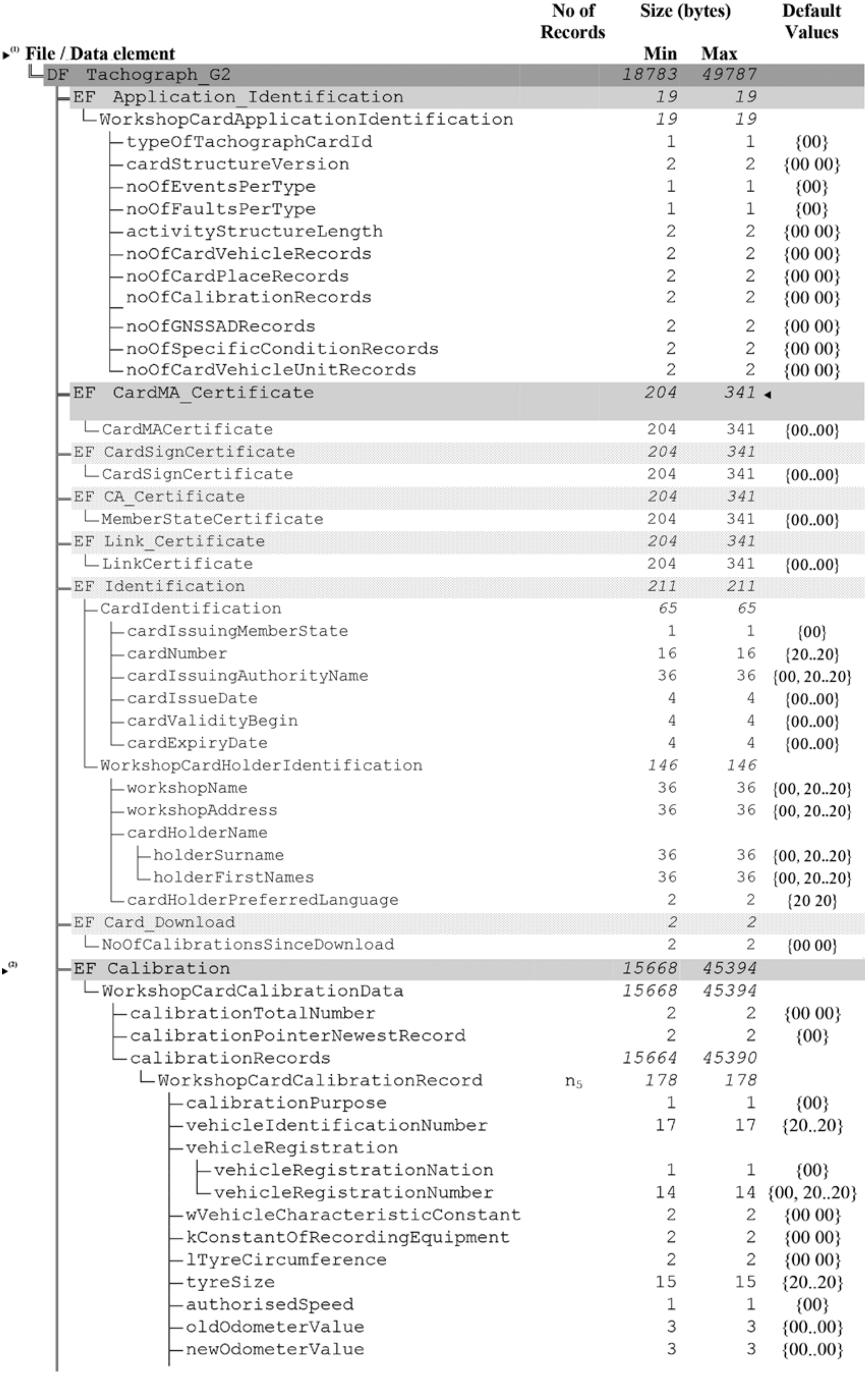
			Access rules		
File	**File ID**	**SFID**	**Read**	**Select**	**Update**
└─DF Tachograph_G2			SC1	SC1	
├─EF Application_Identification	'0501h'	1	SC1	SC1	NEV
├─EF CardMA_Certificate	'C100h'	2	SC1	SC1	NEV
├─EF CardSignCertificate	'C101h'	3	SC1	SC1	NEV
├─EF CA_Certificate	'C108h'	4	SC1	SC1	NEV
├─EF Link_Certificate	'C109h'	5	SC1	SC1	NEV
├─EF Identification	'0520h'	6	SC1	SC1	NEV
├─EF Card_Download	'0509h'	7	SC1	SC1	SC1
├─EF Calibration	'050Ah'	10	SC1	SC1	SM-MAC-G2
├─EF Sensor_Installation_Data	'050Bh'	11	**SC5**	SM-MAC-G2	NEV
├─EF Events_Data	'0502h'	12	SC1	SC1	SM-MAC-G2
├─EF Faults_Data	'0503h'	13	SC1	SC1	SM-MAC-G2
├─EF Driver_Activity_Data	'0504h'	14	SC1	SC1	SM-MAC-G2
├─EF Vehicles_Used	'0505h'	15	SC1	SC1	SM-MAC-G2
├─EF Places	'0506h'	16	SC1	SC1	SM-MAC-G2
├─EF Current_Usage	'0507h'	17	SC1	SC1	SM-MAC-G2
├─EF Control_Activity_Data	'0508h'	18	SC1	SC1	SM-MAC-G2
├─EF Specific_Conditions	'0522h'	19	SC1	SC1	SM-MAC-G2
├─EF VehicleUnits_Used	'0523h'	20	SC1	SC1	SM-MAC-G2
└─EF GNSS_Places	'0524h'	21	SC1	SC1	SM-MAC-G2

The following abbreviations for the security conditions are used in this table:

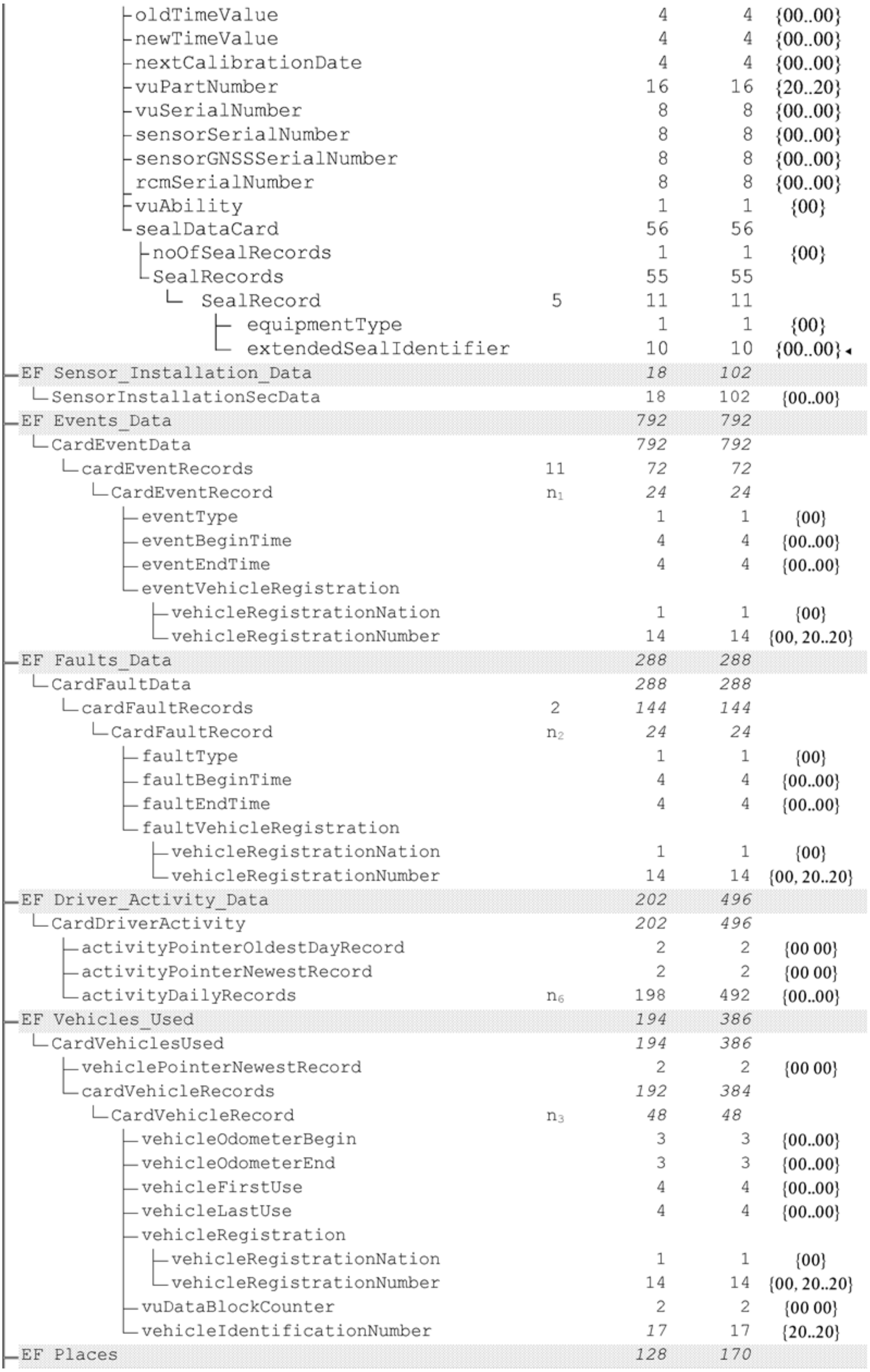
SC1	ALW OR SM-MAC-G2
SC5	For the Read Binary command with even INS byte: SM-C-MAC-G2 AND SM-R-ENC-MAC-G2 For the Read Binary command with odd INS byte (if supported): NEV

TCS_161All EFs structures shall be transparent.

TCS_162The workshop card application generation 2 shall have the following data structure:



▸[1] File / Data element	No of Records	Size (bytes) Min	Size (bytes) Max	Default Values
└DF Tachograph_G2		18783	49787	
─EF Application_Identification		19	19	
└WorkshopCardApplicationIdentification		19	19	
├typeOfTachographCardId		1	1	{00}
├cardStructureVersion		2	2	{00 00}
├noOfEventsPerType		1	1	{00}
├noOfFaultsPerType		1	1	{00}
├activityStructureLength		2	2	{00 00}
├noOfCardVehicleRecords		2	2	{00 00}
├noOfCardPlaceRecords		2	2	{00 00}
├noOfCalibrationRecords		2	2	{00 00}
├noOfGNSSADRecords		2	2	{00 00}
├noOfSpecificConditionRecords		2	2	{00 00}
└noOfCardVehicleUnitRecords		2	2	{00 00}
─EF CardMA_Certificate		204	341 ◂	
└CardMACertificate		204	341	{00..00}
─EF CardSignCertificate		204	341	
└CardSignCertificate		204	341	{00..00}
─EF CA_Certificate		204	341	
└MemberStateCertificate		204	341	{00..00}
─EF Link_Certificate		204	341	
└LinkCertificate		204	341	{00..00}
─EF Identification		211	211	
├CardIdentification		65	65	
├cardIssuingMemberState		1	1	{00}
├cardNumber		16	16	{20..20}
├cardIssuingAuthorityName		36	36	{00, 20..20}
├cardIssueDate		4	4	{00..00}
├cardValidityBegin		4	4	{00..00}
└cardExpiryDate		4	4	{00..00}
└WorkshopCardHolderIdentification		146	146	
├workshopName		36	36	{00, 20..20}
├workshopAddress		36	36	{00, 20..20}
├cardHolderName				
├holderSurname		36	36	{00, 20..20}
└holderFirstNames		36	36	{00, 20..20}
└cardHolderPreferredLanguage		2	2	{20 20}
─EF Card_Download		2	2	
└NoOfCalibrationsSinceDownload		2	2	{00 00}
▸[2] ─EF Calibration		15668	45394	
└WorkshopCardCalibrationData		15668	45394	
├calibrationTotalNumber		2	2	{00 00}
├calibrationPointerNewestRecord		2	2	{00}
└calibrationRecords		15664	45390	
└WorkshopCardCalibrationRecord	n_5	178	178	
├calibrationPurpose		1	1	{00}
├vehicleIdentificationNumber		17	17	{20..20}
├vehicleRegistration				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
├wVehicleCharacteristicConstant		2	2	{00 00}
├kConstantOfRecordingEquipment		2	2	{00 00}
├lTyreCircumference		2	2	{00 00}
├tyreSize		15	15	{20..20}
├authorisedSpeed		1	1	{00}
├oldOdometerValue		3	3	{00..00}
├newOdometerValue		3	3	{00..00}



├oldTimeValue		4	4	{00..00}
├newTimeValue		4	4	{00..00}
├nextCalibrationDate		4	4	{00..00}
├vuPartNumber		16	16	{20..20}
├vuSerialNumber		8	8	{00..00}
├sensorSerialNumber		8	8	{00..00}
├sensorGNSSSerialNumber		8	8	{00..00}
rcmSerialNumber		8	8	{00..00}
├vuAbility		1	1	{00}
└sealDataCard		56	56	
├noOfSealRecords		1	1	{00}
└SealRecords		55	55	
└ SealRecord	5	11	11	
├ equipmentType		1	1	{00}
└ extendedSealIdentifier		10	10	{00..00}◂
EF Sensor_Installation_Data		*18*	*102*	
└SensorInstallationSecData		18	102	{00..00}
EF Events_Data		*792*	*792*	
└CardEventData		*792*	*792*	
└cardEventRecords	11	*72*	*72*	
└CardEventRecord	n_1	*24*	*24*	
├eventType		1	1	{00}
├eventBeginTime		4	4	{00..00}
├eventEndTime		4	4	{00..00}
└eventVehicleRegistration				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
EF Faults_Data		*288*	*288*	
└CardFaultData		*288*	*288*	
└cardFaultRecords	2	*144*	*144*	
└CardFaultRecord	n_2	*24*	*24*	
├faultType		1	1	{00}
├faultBeginTime		4	4	{00..00}
├faultEndTime		4	4	{00..00}
└faultVehicleRegistration				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
EF Driver_Activity_Data		*202*	*496*	
└CardDriverActivity		*202*	*496*	
├activityPointerOldestDayRecord		2	2	{00 00}
├activityPointerNewestRecord		2	2	{00 00}
└activityDailyRecords	n_6	198	492	{00..00}
EF Vehicles_Used		*194*	*386*	
└CardVehiclesUsed		*194*	*386*	
├vehiclePointerNewestRecord		2	2	{00 00}
└cardVehicleRecords		*192*	*384*	
└CardVehicleRecord	n_3	*48*	*48*	
├vehicleOdometerBegin		3	3	{00..00}
├vehicleOdometerEnd		3	3	{00..00}
├vehicleFirstUse		4	4	{00..00}
├vehicleLastUse		4	4	{00..00}
├vehicleRegistration				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
├vuDataBlockCounter		2	2	{00 00}
└vehicleIdentificationNumber		*17*	17	{20..20}
EF Places		*128*	*170*	

└CardPlaceDailyWorkPeriod		*128*	*170*	
├placePointerNewestRecord		2	2	{00 00}
└placeRecords		*126*	*168*	
└PlaceRecord	n_4	*21*	*21*	
├entryTime		4	4	{00..00}
├entryTypeDailyWorkPeriod		1	1	{00}
├dailyWorkPeriodCountry		1	1	{00}
├dailyWorkPeriodRegion		1	1	{00}
├vehicleOdometerValue		3	3	{00..00}
└entryGNSSPlaceRecord		11	11	{00..00}
├timeStamp		4	4	{00..00}
├gnssAccuracy		1	1	{00}
└geoCoordinates		6	6	{00..00}
├EF Current_Usage		*19*	*19*	
└CardCurrentUse		*19*	*19*	
├sessionOpenTime		4	4	{00..00}
└sessionOpenVehicle				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
├EF Control_Activity_Data		*46*	*46*	
└CardControlActivityDataRecord		*46*	*46*	
├controlType		1	1	{00}
├controlTime		4	4	{00..00}
├controlCardNumber				
├cardType		1	1	{00}
├cardIssuingMemberState		1	1	{00}
└cardNumber		16	16	{20..20}
├controlVehicleRegistration				
├vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
├controlDownloadPeriodBegin		4	4	{00..00}
└controlDownloadPeriodEnd		4	4	{00..00}
├EF VehicleUnits_Used		*42*	*42*	
└CardVehicleUnitsUsed		*42*	*82*	
├vehicleUnitPointerNewestRecord		2	2	{00 00}
└cardVehicleUnitRecords		*40*	*80*	
└CardVehicleUnitRecord	n_7	*10*	*10*	
├timeStamp		4	4	{00..00}
├manufacturerCode		1	1	{00..00}
├deviceID		1	1	{00..00}
└vuSoftwareVersion		4	4	{00..00}
▸(1) ├EF GNSS_Places		*326*	*434*	
└GNSSContinuousDriving		*326*	*434*	
├gnssADPointerNewestRecord		2	2	{00 00}
└gnssAccumulatedDrivingRecords		324	432	
└GNSSContinuousDrivingRecord	n_8	*18*	*18*	
├timeStamp		4	4	{00..00}
└gnssPlaceRecord		14	14	
├timeStamp		4	4	{00..00}
├gnssAccuracy		1	1	{00}
├geoCoordinates		6	6	{00..00}
vehicleOdometerValue		3	3	{00..00} ◂
└EF Specific_Conditions		*12*	*22*	
└SpecificConditions		*12*	*22*	
├conditionPointerNewestRecord		2	2	{00 00}
└specificConditionRecords		10	20	
└SpecificConditionRecord	n_9	*5*	*5*	
├entryTime		4	4	{00..00}
└specificConditionType		1	1	{00}

TCS_163The following values, used to provide sizes in the table above, are the minimum and maximum record number values the workshop card data structure must use for a generation 2 application:

		Min	Max
n_1	NoOfEventsPerType	3	3
n_2	NoOfFaultsPerType	6	6
n_3	NoOfCardVehicleRecords	4	8
n_4	NoOfCardPlaceRecords	6	8
n_5	NoOfCalibrationRecords	88	255
n_6	CardActivityLengthRange	198 bytes (1 day * 93 activity changes)	492 bytes (1 day * 240 activity changes)
n_7	NoOfCardVehicleUnitRecords	4	8
▸(f) n_8	NoOfGNSSADRecords	18	24◂
n_9	NoOfSpecificConditionRecords	2	4

4.4. **Control card applications**

4.4.1 *Control Card application generation 1*

TCS_164After its personalisation, the control card application generation 1 shall have the following permanent file structure and file access rules:

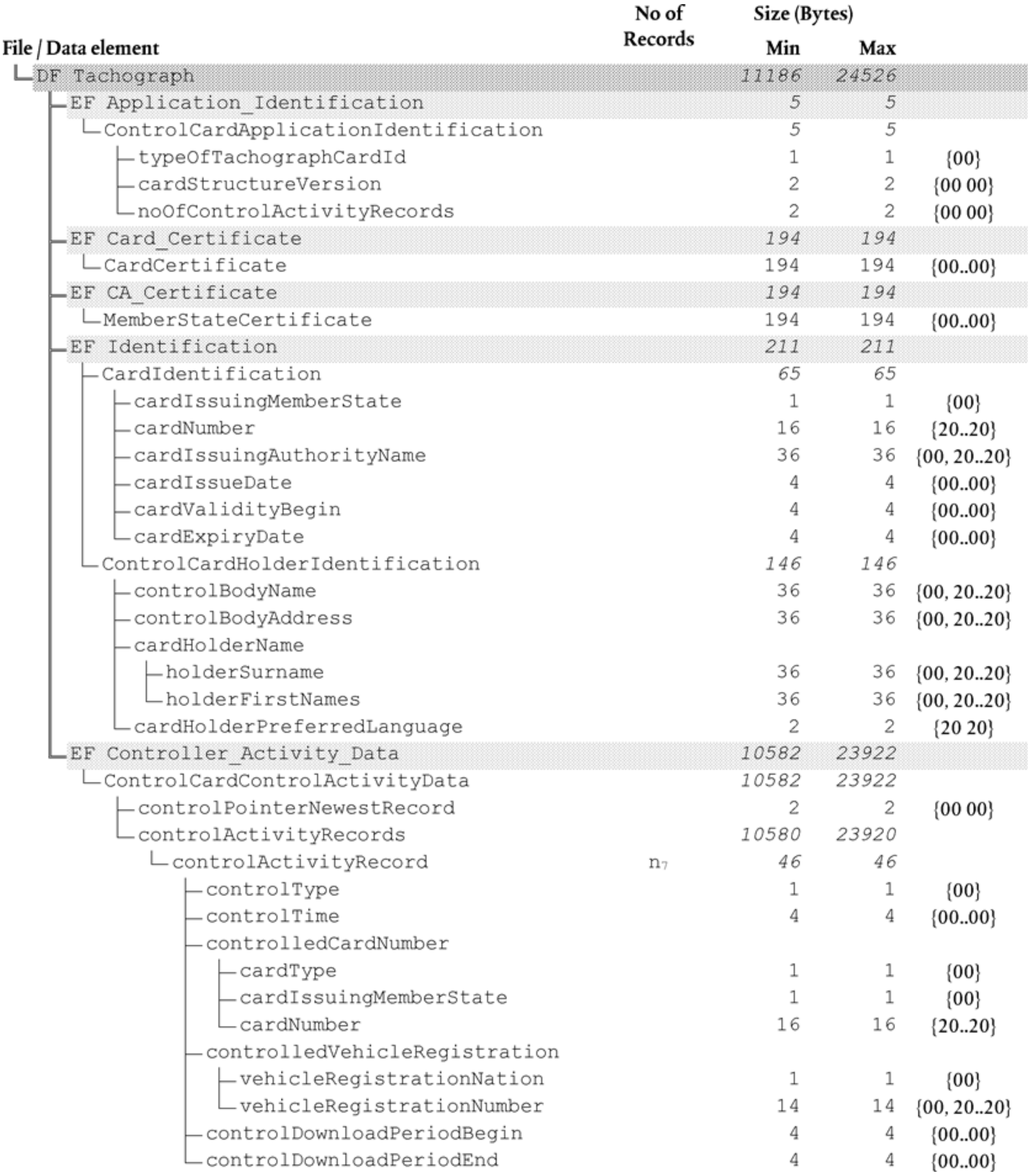
		Access rules		
File	File ID	Read	Select	Update
└─DF Tachograph	'0500h'			
├─EF Application_Identification	'0501h'	SC2	SC1	NEV
├─EF Card_Certificate	'C100h'	SC2	SC1	NEV
├─EF CA_Certificate	'C108h'	SC2	SC1	NEV
├─EF Identification	'0520h'	**SC6**	SC1	NEV
└─EF Controller_Activity_Data	'050Ch'	SC2	SC1	SC3

The following abbreviations for the security conditions are used in this table:

SC1	ALW OR SM-MAC-G2
SC2	ALW OR SM-MAC-G1 OR SM-MAC-G2
SC3	SM-MAC-G1 OR SM-MAC-G2
SC6	EXT-AUT-G1 OR SM-MAC-G1 OR SM-MAC-G2

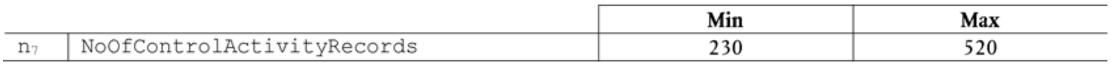
TCS_165All EF structures shall be transparent.

TCS_166The control card application generation 1 shall have the following data structure:



File / Data element	No of Records	Size (Bytes)		
		Min	Max	
└DF Tachograph		*11186*	*24526*	
├EF Application_Identification		*5*	*5*	
│└ControlCardApplicationIdentification		*5*	*5*	
│ ├typeOfTachographCardId		1	1	{00}
│ ├cardStructureVersion		2	2	{00 00}
│ └noOfControlActivityRecords		2	2	{00 00}
├EF Card_Certificate		*194*	*194*	
│└CardCertificate		194	194	{00..00}
├EF CA_Certificate		*194*	*194*	
│└MemberStateCertificate		194	194	{00..00}
├EF Identification		*211*	*211*	
│├CardIdentification		*65*	*65*	
││├cardIssuingMemberState		1	1	{00}
││├cardNumber		16	16	{20..20}
││├cardIssuingAuthorityName		36	36	{00, 20..20}
││├cardIssueDate		4	4	{00..00}
││├cardValidityBegin		4	4	{00..00}
││└cardExpiryDate		4	4	{00..00}
│└ControlCardHolderIdentification		*146*	*146*	
│ ├controlBodyName		36	36	{00, 20..20}
│ ├controlBodyAddress		36	36	{00, 20..20}
│ ├cardHolderName				
│ │├holderSurname		36	36	{00, 20..20}
│ │└holderFirstNames		36	36	{00, 20..20}
│ └cardHolderPreferredLanguage		2	2	{20 20}
└EF Controller_Activity_Data		*10582*	*23922*	
└ControlCardControlActivityData		*10582*	*23922*	
├controlPointerNewestRecord		2	2	{00 00}
└controlActivityRecords		*10580*	*23920*	
└controlActivityRecord	n_7	*46*	*46*	
├controlType		1	1	{00}
├controlTime		4	4	{00..00}
├controlledCardNumber				
│├cardType		1	1	{00}
│├cardIssuingMemberState		1	1	{00}
│└cardNumber		16	16	{20..20}
├controlledVehicleRegistration				
│├vehicleRegistrationNation		1	1	{00}
│└vehicleRegistrationNumber		14	14	{00, 20..20}
├controlDownloadPeriodBegin		4	4	{00..00}
└controlDownloadPeriodEnd		4	4	{00..00}

TCS_167The following values, used to provide sizes in the table above, are the minimum and maximum record number values the control card data structure must use for a generation 1 application:

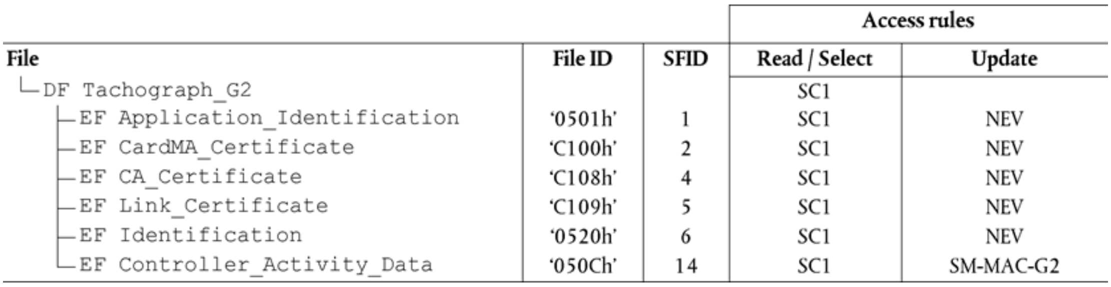


		Min	Max
n_7	NoOfControlActivityRecords	230	520

4.4.2 *Control card application generation 2*

TCS_168After its personalisation, the control card application generation 2 shall have the following permanent file structure and file access rules.

Note: The short EF identifier SFID is given as decimal number, e.g. the value 30 corresponds to 11110 in binary.



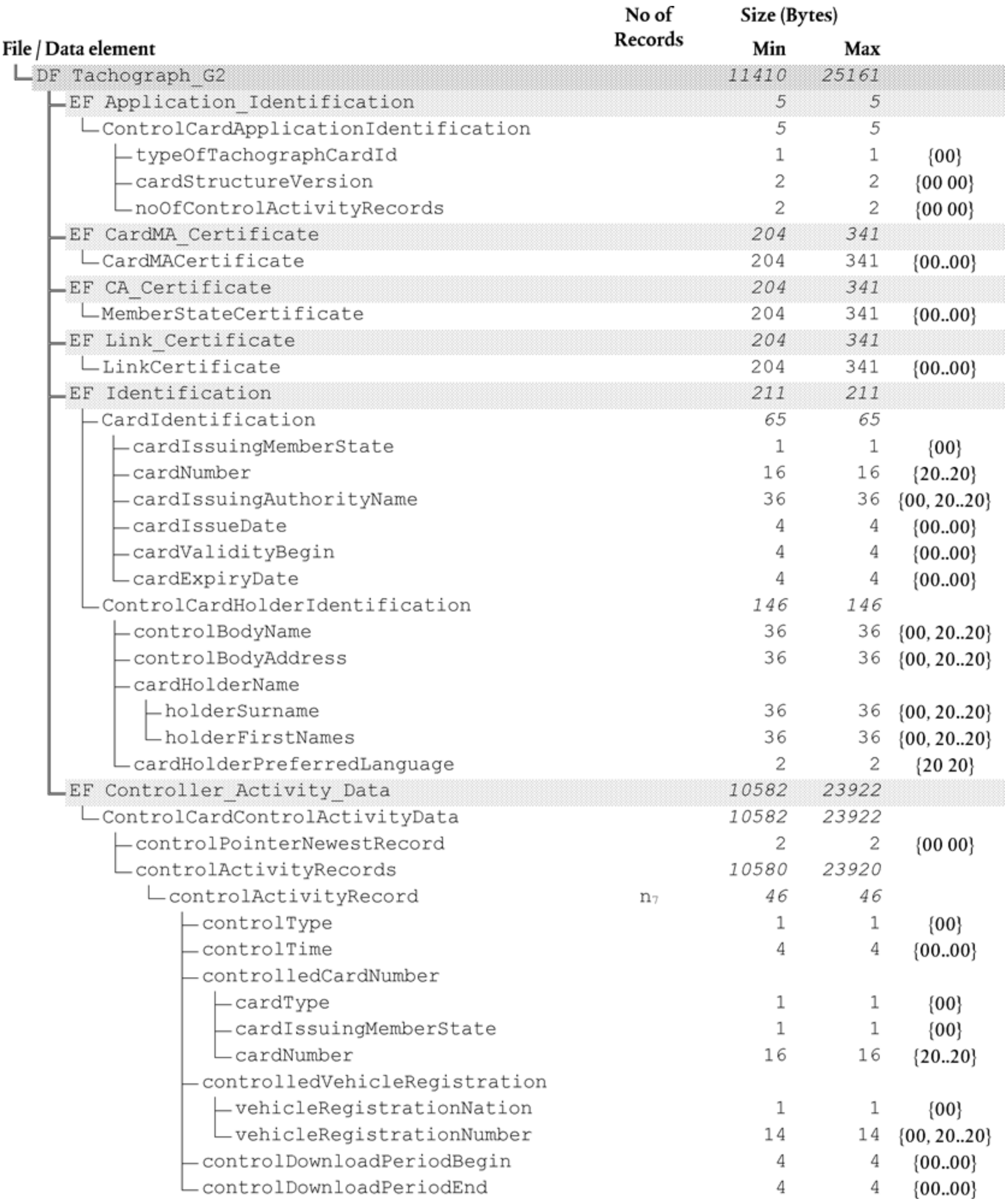
			Access rules	
File	**File ID**	**SFID**	**Read / Select**	**Update**
└─DF Tachograph_G2			SC1	
├─EF Application_Identification	'0501h'	1	SC1	NEV
├─EF CardMA_Certificate	'C100h'	2	SC1	NEV
├─EF CA_Certificate	'C108h'	4	SC1	NEV
├─EF Link_Certificate	'C109h'	5	SC1	NEV
├─EF Identification	'0520h'	6	SC1	NEV
└─EF Controller_Activity_Data	'050Ch'	14	SC1	SM-MAC-G2

The following abbreviation for the security condition is used in this table:

SC1 ALW OR SM-MAC-G2

TCS_169All EF structures shall be transparent.

TCS_170The control card application generation2 shall have the following data structure:



File / Data element	No of Records	Size (Bytes)		
		Min	Max	
└DF Tachograph_G2		*11410*	*25161*	
├EF Application_Identification		*5*	*5*	
│└ControlCardApplicationIdentification		*5*	*5*	
│ ├typeOfTachographCardId		1	1	{00}
│ ├cardStructureVersion		2	2	{00 00}
│ └noOfControlActivityRecords		2	2	{00 00}
├EF CardMA_Certificate		*204*	*341*	
│└CardMACertificate		204	341	{00..00}
├EF CA_Certificate		*204*	*341*	
│└MemberStateCertificate		204	341	{00..00}
├EF Link_Certificate		*204*	*341*	
│└LinkCertificate		204	341	{00..00}
├EF Identification		*211*	*211*	
│├CardIdentification		*65*	*65*	
││├cardIssuingMemberState		1	1	{00}
││├cardNumber		16	16	{20..20}
││├cardIssuingAuthorityName		36	36	{00, 20..20}
││├cardIssueDate		4	4	{00..00}
││├cardValidityBegin		4	4	{00..00}
││└cardExpiryDate		4	4	{00..00}
│└ControlCardHolderIdentification		*146*	*146*	
│ ├controlBodyName		36	36	{00, 20..20}
│ ├controlBodyAddress		36	36	{00, 20..20}
│ ├cardHolderName				
│ │├holderSurname		36	36	{00, 20..20}
│ │└holderFirstNames		36	36	{00, 20..20}
│ └cardHolderPreferredLanguage		2	2	{20 20}
└EF Controller_Activity_Data		*10582*	*23922*	
└ControlCardControlActivityData		*10582*	*23922*	
├controlPointerNewestRecord		2	2	{00 00}
└controlActivityRecords		*10580*	*23920*	
└controlActivityRecord	n_7	*46*	*46*	
├controlType		1	1	{00}
├controlTime		4	4	{00..00}
├controlledCardNumber				
│├cardType		1	1	{00}
│├cardIssuingMemberState		1	1	{00}
│└cardNumber		16	16	{20..20}
├controlledVehicleRegistration				
│├vehicleRegistrationNation		1	1	{00}
│└vehicleRegistrationNumber		14	14	{00, 20..20}
├controlDownloadPeriodBegin		4	4	{00..00}
└controlDownloadPeriodEnd		4	4	{00..00}

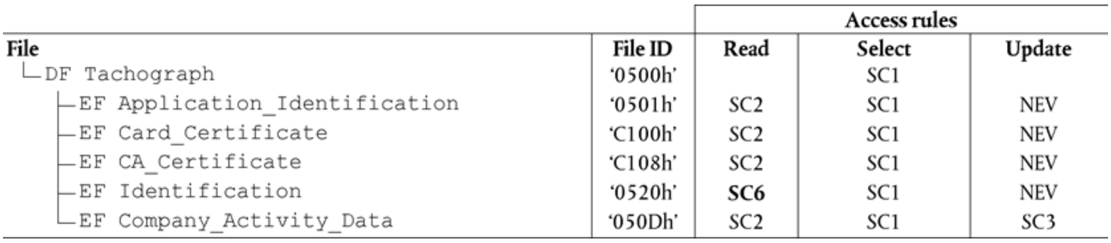
TCS_171 The following values, used to provide sizes in the table above, are the minimum and maximum record number values the control card data structure must use for a generation 2 application:

		Min	Max
n_7	NoOfControlActivityRecords	230	520

4.5. Company card applications

4.5.1 *Company card application generation 1*

TCS_172After its personalisation, the company card application generation 1 shall have the following permanent file structure and file access rules:



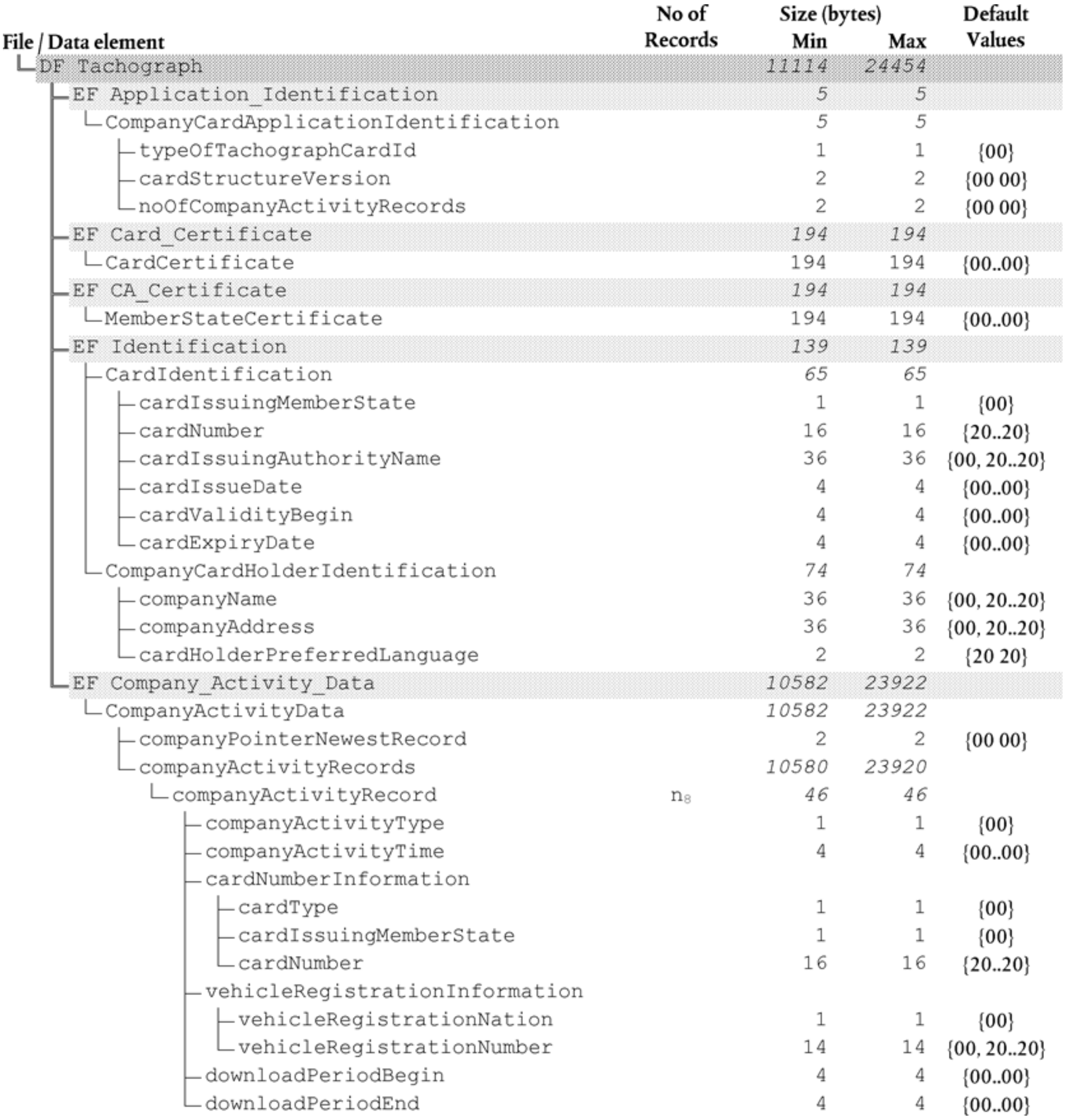
		Access rules		
File	**File ID**	**Read**	**Select**	**Update**
└DF Tachograph	'0500h'		SC1	
├EF Application_Identification	'0501h'	SC2	SC1	NEV
├EF Card_Certificate	'C100h'	SC2	SC1	NEV
├EF CA_Certificate	'C108h'	SC2	SC1	NEV
├EF Identification	'0520h'	**SC6**	SC1	NEV
└EF Company_Activity_Data	'050Dh'	SC2	SC1	SC3

The following abbreviations for the security conditions are used in this table:

SC1	ALW OR SM-MAC-G2
SC2	ALW OR SM-MAC-G1 OR SM-MAC-G2
SC3	SM-MAC-G1 OR SM-MAC-G2
SC6	EXT-AUT-G1 OR SM-MAC-G1 OR SM-MAC-G2

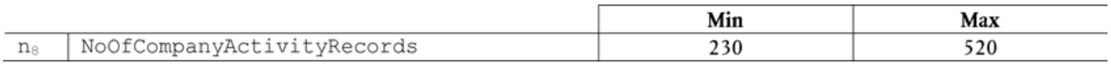
TCS_173All EF structures shall be transparent.

TCS_174The company card application generation 1 shall have the following data structure:



File / Data element	No of Records	Size (bytes) Min	Size (bytes) Max	Default Values
└DF Tachograph		*11114*	*24454*	
├EF Application_Identification		*5*	*5*	
│ └CompanyCardApplicationIdentification		*5*	*5*	
│ │ ├typeOfTachographCardId		1	1	{00}
│ │ ├cardStructureVersion		2	2	{00 00}
│ │ └noOfCompanyActivityRecords		2	2	{00 00}
├EF Card_Certificate		*194*	*194*	
│ └CardCertificate		194	194	{00..00}
├EF CA_Certificate		*194*	*194*	
│ └MemberStateCertificate		194	194	{00..00}
├EF Identification		*139*	*139*	
│ ├CardIdentification		*65*	*65*	
│ │ ├cardIssuingMemberState		1	1	{00}
│ │ ├cardNumber		16	16	{20..20}
│ │ ├cardIssuingAuthorityName		36	36	{00, 20..20}
│ │ ├cardIssueDate		4	4	{00..00}
│ │ ├cardValidityBegin		4	4	{00..00}
│ │ └cardExpiryDate		4	4	{00..00}
│ └CompanyCardHolderIdentification		*74*	*74*	
│ │ ├companyName		36	36	{00, 20..20}
│ │ ├companyAddress		36	36	{00, 20..20}
│ │ └cardHolderPreferredLanguage		2	2	{20 20}
└EF Company_Activity_Data		*10582*	*23922*	
│ └CompanyActivityData		*10582*	*23922*	
│ │ ├companyPointerNewestRecord		2	2	{00 00}
│ │ └companyActivityRecords		*10580*	*23920*	
│ │ │ └companyActivityRecord	n_8	*46*	*46*	
│ │ │ │ ├companyActivityType		1	1	{00}
│ │ │ │ ├companyActivityTime		4	4	{00..00}
│ │ │ │ ├cardNumberInformation				
│ │ │ │ │ ├cardType		1	1	{00}
│ │ │ │ │ ├cardIssuingMemberState		1	1	{00}
│ │ │ │ │ └cardNumber		16	16	{20..20}
│ │ │ │ ├vehicleRegistrationInformation				
│ │ │ │ │ ├vehicleRegistrationNation		1	1	{00}
│ │ │ │ │ └vehicleRegistrationNumber		14	14	{00, 20..20}
│ │ │ │ ├downloadPeriodBegin		4	4	{00..00}
│ │ │ │ └downloadPeriodEnd		4	4	{00..00}

TCS_175The following values, used to provide sizes in the table above, are the minimum and maximum record number values the company card data structure must use for a generation 1 application:

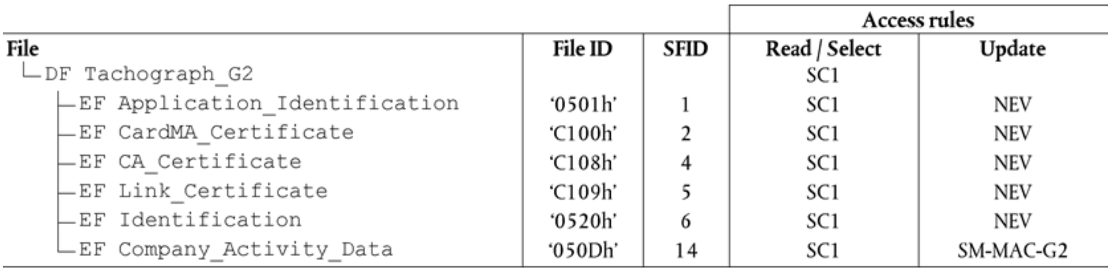


		Min	Max
n_8	NoOfCompanyActivityRecords	230	520

4.5.2 *Company card application generation 2*

TCS_176After its personalisation, the company card application generation 2 shall have the following permanent file structure and file access rules.

Note: The short EF identifier SFID is given as decimal number, e.g. the value 30 corresponds to 11110 in binary.



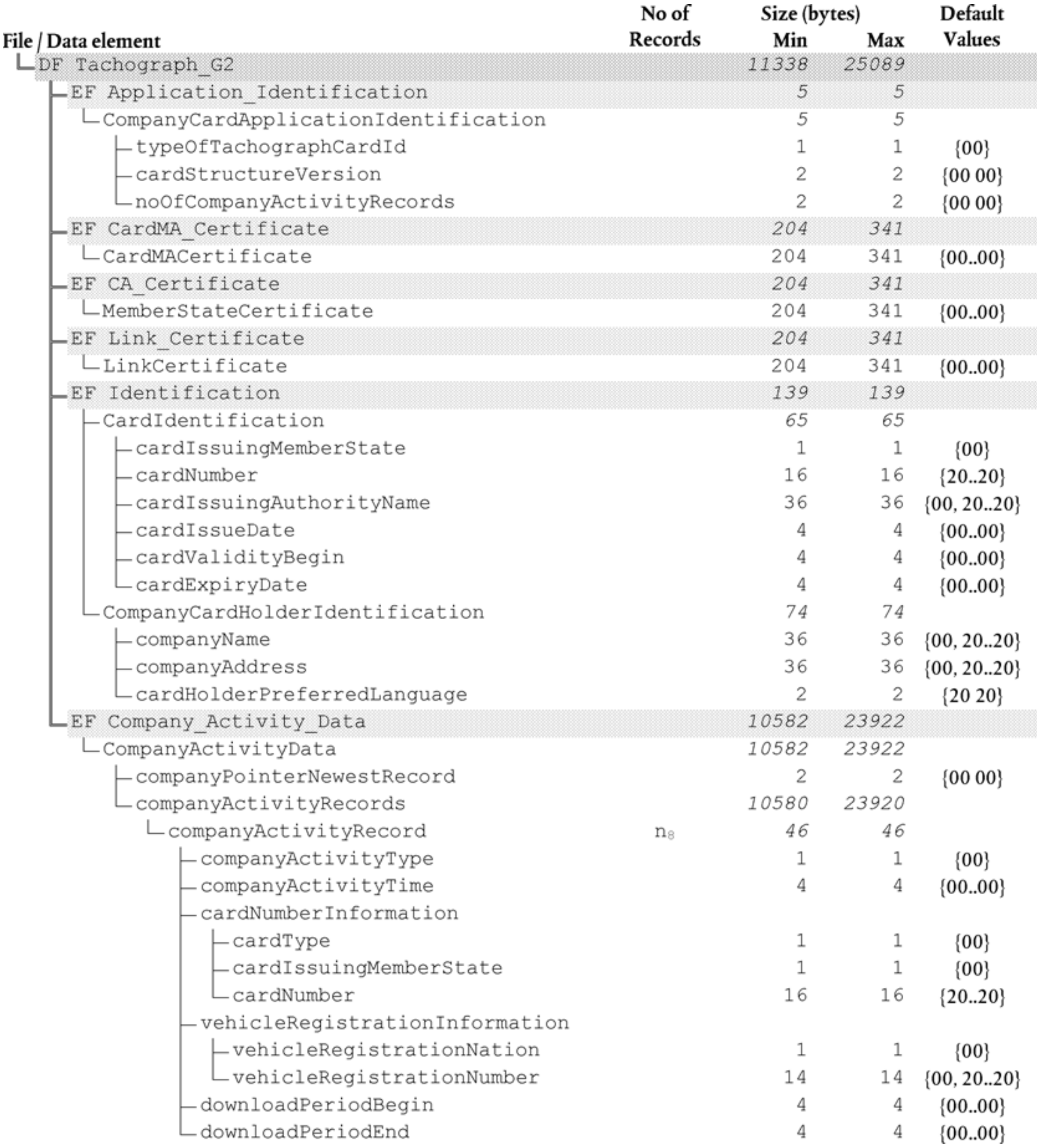
			Access rules	
File	**File ID**	**SFID**	**Read / Select**	**Update**
└DF Tachograph_G2			SC1	
├EF Application_Identification	'0501h'	1	SC1	NEV
├EF CardMA_Certificate	'C100h'	2	SC1	NEV
├EF CA_Certificate	'C108h'	4	SC1	NEV
├EF Link_Certificate	'C109h'	5	SC1	NEV
├EF Identification	'0520h'	6	SC1	NEV
└EF Company_Activity_Data	'050Dh'	14	SC1	SM-MAC-G2

The following abbreviation for the security condition is used in this table:

SC1	ALW OR SM-MAC-G2

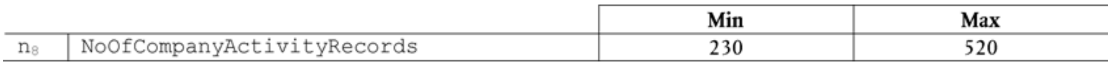
TCS_177All EF structures shall be transparent.

TCS_178The company card application generation 2 shall have the following data structure:



File / Data element	No of Records	Size (bytes) Min	Size (bytes) Max	Default Values
└DF Tachograph_G2		11338	25089	
─EF Application_Identification		5	5	
└CompanyCardApplicationIdentification		5	5	
─typeOfTachographCardId		1	1	{00}
─cardStructureVersion		2	2	{00 00}
└noOfCompanyActivityRecords		2	2	{00 00}
─EF CardMA_Certificate		204	341	
└CardMACertificate		204	341	{00..00}
─EF CA_Certificate		204	341	
└MemberStateCertificate		204	341	{00..00}
─EF Link_Certificate		204	341	
└LinkCertificate		204	341	{00..00}
─EF Identification		139	139	
─CardIdentification		65	65	
─cardIssuingMemberState		1	1	{00}
─cardNumber		16	16	{20..20}
─cardIssuingAuthorityName		36	36	{00, 20..20}
─cardIssueDate		4	4	{00..00}
─cardValidityBegin		4	4	{00..00}
└cardExpiryDate		4	4	{00..00}
└CompanyCardHolderIdentification		74	74	
─companyName		36	36	{00, 20..20}
─companyAddress		36	36	{00, 20..20}
└cardHolderPreferredLanguage		2	2	{20 20}
└EF Company_Activity_Data		10582	23922	
└CompanyActivityData		10582	23922	
─companyPointerNewestRecord		2	2	{00 00}
└companyActivityRecords		10580	23920	
└companyActivityRecord	n_8	46	46	
─companyActivityType		1	1	{00}
─companyActivityTime		4	4	{00..00}
─cardNumberInformation				
─cardType		1	1	{00}
─cardIssuingMemberState		1	1	{00}
└cardNumber		16	16	{20..20}
─vehicleRegistrationInformation				
─vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
─downloadPeriodBegin		4	4	{00..00}
└downloadPeriodEnd		4	4	{00..00}

TCS_179 The following values, used to provide sizes in the table above, are the minimum and maximum record number values the company card data structure must use for a generation 2 application:



		Min	Max
n_8	NoOfCompanyActivityRecords	230	520