ETSI TS 102 226 V16.1.0 (2022-10)

Smart Cards;
Remote APDU structure for UICC based applications
(Release 16)

Reference
RTS/SET-T102226vg10

Keywords
protocol, smart card

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Secure Element Technologies (SET).

It is based on work originally done in the 3GPP in TSG-terminals WG3 and ETSI SMG.

The contents of the present document are subject to continuing work within TC SET and may change following formal TC SET approval. If TC SET modifies the contents of the present document, it will then be republished by ETSI with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x: the first digit:
  - 0 early working draft;
  - 1 presented to TC SET for information;
  - 2 presented to TC SET for approval;
  - 3 or greater indicates TC SET approved document under change control.
- y: the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z: the third digit is incremented when editorial only changes have been incorporated in the document.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document defines the remote management of the UICC based on any of the secured packet structures specified in ETSI TS 102 225 [1].

It specifies the APDU format for remote management.

Furthermore the present document specifies:

- A set of commands coded according to this APDU structure and used in the remote file management on the UICC. This is based on ETSI TS 102 221 [2].
- A set of commands coded according to this APDU structure and used in the remote application management on the UICC. This is based on the GlobalPlatform Card Specifications.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

- In the case of a reference to a TC SET document, a non-specific reference implicitly refers to the latest version of that document in the same Release as the present document.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] ETSI TS 102 225: "Smart Cards; Secured packet structure for UICC based applications".

[2] ETSI TS 102 221: "Smart Cards; UICC-Terminal interface; Physical and logical characteristics".

[3] ETSI TS 102 223: "Smart Cards; Card Application Toolkit (CAT)".

[4] GlobalPlatform: "GlobalPlatform Card Specification Version 2.3".

NOTE: See http://www.globalplatform.org/.

[5] ETSI TS 101 220: "Smart Cards; ETSI numbering system for telecommunication application providers".

[6] ETSI TS 102 241: "Smart Cards; UICC Application Programming Interface (UICC API) for Java Card™".

[7] Void.

[8] Void.

[9] ETSI TS 102 222: "Integrated Circuit Cards (ICC); Administrative commands for telecommunications applications".

[10] ETSI TS 123 048: "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); Security mechanisms for the (U)SIM application toolkit; Stage 2 (3GPP TS 23.048 Release 5)".

[11] ETSI TS 102 127: "Smart Cards; Transport protocol for CAT applications; Stage 2".

[12] ETSI TS 143 019: "Digital cellular telecommunications system (Phase 2+); Subscriber Identity Module Application Programming Interface (SIM API) for Java Card; Stage 2 (3GPP TS 43.019 Release 5)".

[13] FIPS-197 (2001): "Advanced Encryption Standard (AES)".

NOTE: Available at https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.197.pdf.

[14] NIST Special Publication 800-38A (2001): "Recommendation for Block Cipher Modes of Operation - Methods and Techniques".

NOTE: Available at http://csrc.nist.gov/publications/nistpubs/.

[15] NIST Special Publication 800-38B (2001): "Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication".

NOTE: Available at http://csrc.nist.gov/publications/nistpubs/.

[16] GlobalPlatform: "Card UICC Configuration", Version 2.0.

NOTE: Available at http://www.globalplatform.org/.

[17] ETSI TS 102 588: "Smart Cards; Application invocation Application Programming Interface (API) by a UICC webserver for Java Card™ platform".

[18] GlobalPlatform: "GlobalPlatform Card, Confidential Card Content Management Card Specification v2.3 - Amendment A", Version 1.1.

NOTE: Available at http://www.globalplatform.org/.

[19] GlobalPlatform: "GlobalPlatform Card, Remote Application Management over HTTP, Card Specification v2.2, Amendment B" Version 1.1.3.

NOTE: Available at http://www.globalplatform.org/.

[20] ETSI TS 102 483: "Smart cards; UICC-Terminal interface; Internet Protocol connectivity between UICC and terminal".

[21] ISO/IEC 8825-1: "Information technology - ASN.1 encoding rules - Part 1: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)".

[22] GlobalPlatform: "Card Specification Version 2.3, Amendment C: Contactless Services" Version 1.2.

NOTE: Available at http://www.globalplatform.org/.

[23] ETSI TS 102 622: "Smart Card; UICC - Contactless Front-end (CLF) Interface; Host Controller Interface (HCI)".

[24] GlobalPlatform: "Security Upgrade for Card Content Management - GlobalPlatform Card Specification v2.2 - Amendment E", Version 1.0.1.

NOTE: Available at http://www.globalplatform.org/.

[25] GlobalPlatform: "Java Card API and Export File for Card Specification v2.2.1 (org.globalplatform) Version 1.6".

NOTE: Available at http://www.globalplatform.org/.

[26] GlobalPlatform: "Card Specification Version 2.2 - Amendment D: Secure Channel Protocol 03" Version 1.1.1.

NOTE: Available at http://www.globalplatform.org/.

[27] GlobalPlatform: "GlobalPlatform Card, Common Implementation Configuration", Version 2.0.

NOTE: Available at http://www.globalplatform.org/.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

- In the case of a reference to a TC SET document, a non-specific reference implicitly refers to the latest version of that document in the same Release as the present document.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

Not applicable.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI TS 102 225 [1], ETSI TS 101 220 [5] and the following apply:

Controlling Authority Security Domain (CASD): security domain providing cryptographic functions, as specified in GlobalPlatform Card Specification Amendment A [18]

NOTE: It provides services to confidentially load or generate Secure Channel keys of an APSD.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI TS 102 225 [1] and the following apply:

ACK	ACKnowledge
ADD	Access Domain Data
ADF	Application Data File
ADP	Access Domain Parameter
AES	Advanced Encryption Standard
AFI	Application Family Identifier
AID	Application IDentifier
AM	Authorized Management
AP	Application Provider
APDU	Application Protocol Data Unit
API	Application Programming Interface
APSD	Application Provider Security Domain
BER-TLV	Basic Encoding Rules - Tag, Length, Value
BIP	Bearer Independent Protocol
C-APDU	Command Application Protocol Data Unit
CASD	Controlling Authority Security Domain

CAT_TP	Card Application Toolkit Transport Protocol
CBC	Cell Broadcast Centre
CC	Cryptographie Checksum
CL	ContactLess
CLA	Class
CLT	Contactless Tunneling
CMAC	Cipher-based Message Authentication Code
DAP	Data Authentication Pattern
DEK	Data Encryption Key
DES	Data Encryption Standard
DF	Directory File
DM	Delegated Management
DS	Digital Signature
ECB	Electronic Code Book
ECKA	Elliptic Curve Key Agreement algorithm
ECKA-EG	ElGamal ECKA
EF	Elementary File
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
ICCID	Integrated Circuit Card IDentification
ICV	Integrity Check Value
INS	INStruction
IP	Internet Protocol
ISD	Issuer Security Domain
KIc	Key and algorithm Identifier for ciphering
KID	Key and algorithm IDentifier for RC/CC/DS
MAC	Message Authentication Code
MF	Management Field
MSL	Minimum Security Level
MSLD	Minimum Security Level Data
NIST	National Institute of Standards and Technology
OTA	Over The Air
PDU	Packet Data Unit
PIN	Personal Identification Number
RAM	Remote Application Management
R-APDU	Response Application Protocol Data Unit
RF	Radio Frequency
RFM	Remote File Management
RFU	Reserved for Future Use
SCP02	Secure Channel Protocol 02
SCP03	Secure Channel Protocol 03
SD	Security Domain
SDU	Service Data Unit
SE	Sending Entity
SMG	Special Mobile Group
SP	Special Publication
SPI	Security Parameter Indication
TAR	Toolkit Application Reference
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TLV	Tag Length Value
TPDU	Transfer Protocol Data Unit

4 Overview of remote management

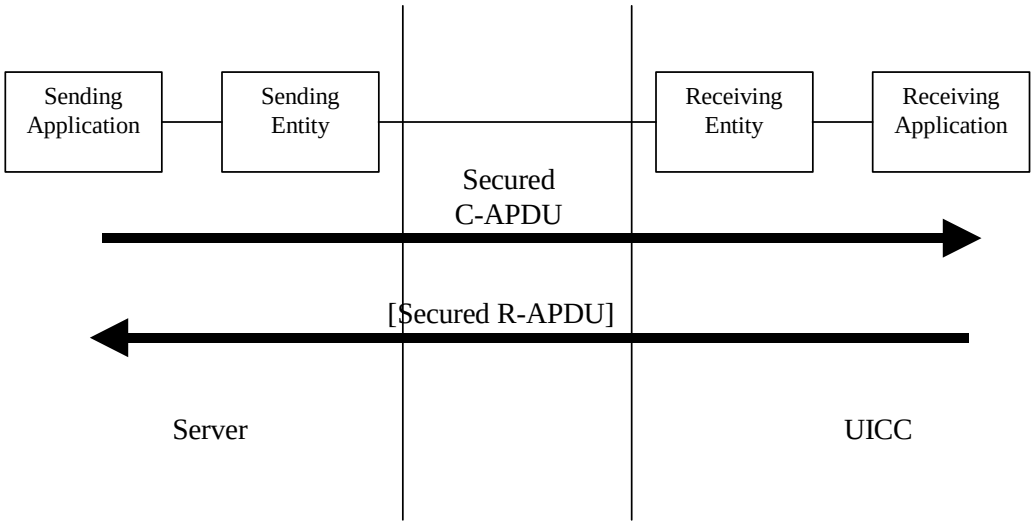

Figure 4.1: Remote management

All data exchanged between the Sending Entity and Receiving Entity shall be formatted as "Secured data" according to ETSI TS 102 225 [1]:

1) The parameter(s) (the command string) in the "Secured data" is either a single command, or a list of commands, which shall be processed sequentially. Additional application provider security may be applied to the "secured data" as specified in clause 10.2 of the present document.

2) The Remote Management application shall take parameters from the "Secured data" and shall act upon the files or applications or perform other actions according to these parameters. A Remote Management application is the on-card Receiving Application that performs either Remote File Management (RFM) or Remote Application Management (RAM) as defined in the following clauses.

3) Remote Management commands shall be executed by the dedicated Remote Management Application. A "Command session" is defined as starting upon receipt of the parameter/command list, and ends when the parameter list in the "Secured data" is completed, or when an error (i.e. SW1 of the command indicates an error condition) is detected which shall halt further processing of the command list. Warnings or procedure bytes do not halt processing of the command list. Such a "Command session" shall be handled like an application session defined in ETSI TS 102 221 [2] (for RFM) and GlobalPlatform Card Specification [4] (for RAM). Application selection at the beginning of the session happens implicitly based on the header information (TAR or HTTP header field X-Admin-Targeted-Application). Unless defined otherwise in the present document, the session context shall be deleted when the "Command session" ends.

4) At the beginning and end of a Command "session" the logical state of the UICC as seen from the terminal shall not be changed to an extent sufficient to disrupt the behaviour of the terminal. If changes in the logical state have occurred that the terminal needs to be aware of, the application on the UICC may issue a REFRESH command according to ETSI TS 102 223 [3].

The processing of the security in the Receiving Entity according to ETSI TS 102 225 [1] and according to the present document is one of the tasks of a Security Domain according to GlobalPlatform Card Specification [4].

The mechanism defined above (addressing and selection based on TAR or HTTP header field X-Admin-Targeted-Application) can also be used to send data to a Receiving Application which is not a Remote Management Application. In this case the format of the data exchanged between Sending Application and Receiving Application is application specific and not defined in the present document.

5 Remote APDU format

5.1 Compact Remote Application data format

5.1.1 Compact Remote command structure

A command string may contain a single command or a sequence of commands. The structure of each command shall be according to the generalized structure defined below; each element other than the Data field is a single octet (see ETSI TS 102 221 [2]).

The format of the commands is the same as the one defined in ETSI TS 102 221 [2] for T = 0 TPDU commands.

Class byte (CLA)	Instruction code (INS)	P1	P2	P3	Data

If the sending application needs to retrieve the Response parameters/data of a case 4 command, then a GET RESPONSE command shall follow this command in the command string.

The GET RESPONSE and any case 2 command (i.e. READ BINARY, READ RECORD) shall only occur once in a command string and, if present, shall be the last command in the string.

For all case 2 commands and for the GET RESPONSE command, if P3 = '00', then the UICC shall send back all available response parameters/data e.g. if a READ RECORD command has P3 = '00' the whole record shall be returned. The limitation of 256 bytes does not apply for the length of the response data. In case the data is truncated in the response, the remaining bytes are lost and the status words shall be set to '62 F1'.

5.1.2 Compact Remote response structure

If a proof of Receipt is required by the sending entity, the Additional Response Data sent by the Remote Management Application shall be formatted according to table 5.1.

Table 5.1: Format of additional response data

Length	Name
1	Number of commands executed within the command script (see note)
2	Status bytes or '61 xx' procedure bytes of last executed command/GET RESPONSE
X	Response data of last executed command/GET RESPONSE if available (i.e. if the last command was a case 2 command or a GET RESPONSE)
NOTE:	This field shall be set to '01' if one command was executed within the command script, '02' if two commands were executed, etc.

5.2 Expanded Remote Application data format

5.2.1 Expanded Remote command structure

5.2.1.0 Structure overview

The "Secured data" sent to a Remote Management Application shall be a BER-TLV data object formatted according to table 5.2.

Two variants exist for the expanded remote command structure:

- The Command Scripting template is a BER-TLV data object as defined in ETSI TS 101 220 [5], i.e. it uses definite length coding; see table 5.2.
- The Command Scripting template is a BER-TLV data object which uses indefinite length coding as defined in ISO/IEC 8825-1 [21]; see table 5.2a.

NOTE: The variant with indefinite length coding is recommended to be used for RAM/RFM over HTTPS.

The tags of these TLVs are defined in annex A.

Table 5.2: Expanded format of Remote Management application command "secured data" - definite length coding

Length in bytes	Name
1	Command Scripting template tag for definite length coding
L	Length of Command Scripting template= A+B+…C
A	Command TLV
B	Command TLV
	…
C	Command TLV

Table 5.2a: Expanded format of Remote Management application command "secured data" - indefinite length coding

Length in bytes	Name
1	Command Scripting template tag for indefinite length coding
1	Indicator for indefinite length coding (value '80')
A	Command TLV
B	Command TLV
	…
C	Command TLV
2	End of content indicator (value '00 00')

A Remote Management application command string may contain a single or several Command TLVs.

A Command TLV can be one of the following:

- A C-APDU, containing a remote management command.
- An Immediate Action TLV, containing a proactive command or another action to be performed when it is encountered while processing the sequence of Command TLVs.
- An Error Action TLV, containing a proactive command to be performed only if an error is encountered in a C-APDU following this TLV.
- A script Chaining TLV as first Command TLV.

5.2.1.1 C-APDU TLV

The structure of each C-APDU shall be a TLV structure coded according to the C-APDU COMPREHENSION-TLV data object coding defined in ETSI TS 102 223 [3]. The restriction on the length of the C-APDU mentioned in the note in ETSI TS 102 223 [3] shall not apply.

For all case 2 and case 4 C-APDUs, if Le='00' in the C-APDU, then the UICC shall send back all available response parameters/data in the R-APDU e.g. if a READ RECORD command has Le='00' the whole record shall be returned. The limitation of 256 bytes does not apply for the length of the response data.

In case the data is truncated in the response of a C-APDU, the status words for this C-APDU shall be set to '62 F1' in the corresponding R-APDU. This shall terminate the processing of the command list.

If a R-APDU fills the response buffer so that no further R-APDU can be included in the response scripting template, this shall terminate the processing of the command list.

If Le field is empty in the C-APDU, then no response data is expected in the R-APDU and in case of expanded format with definite length coding, no R-APDU shall be returned by the UICC in the application additional response data except if the corresponding C-APDU is the last command executed in the script.

NOTE: In this expanded format the GET RESPONSE command is not used.

5.2.1.2 Immediate Action TLV

The Immediate Action TLV is a BER-TLV data object that allows the Remote Management Application to issue a proactive command during the execution or that allows to abort the execution if a proactive session is ongoing. It shall be formatted as shown in table 5.3 or table 5.4.

Table 5.3: Immediate Action TLV - normal format

Length in bytes	Name
1	Immediate Action tag (see annex A)
L	Length of Immediate Action = A > 1
A	Set of COMPREHENSION-TLV data objects

Table 5.4: Immediate Action TLV - referenced format

Length in bytes	Name
1	Immediate Action tag (see annex A)
1	Length of Immediate Action = 1
1	'01' to '7F': Reference to a record in EF_{RMA} '81': Proactive session indication '82': Early response other values: RFU

In case of reference to a record in EF_{RMA}, the referenced record shall contain the set of COMPREHENSION-TLV data objects preceded by a length value as defined for a BER-TLV, see ETSI TS 102 222 [9].

If present, the Immediate Action TLV coding "proactive session indication" shall be:

- The first Command TLV in the script if there is no script chaining.
- The second Command TLV in the script if there is script chaining.

In case of "proactive session indication", execution of the remaining script shall be suspended if a proactive session is ongoing. Script processing shall be resumed after the end of the proactive session. If the UICC cannot suspend the script execution, e.g. because there is not enough internal resources available, the UICC shall terminate the processing of the script and return a "suspension error" in the response data.

If no "proactive session indication" is present as first Command TLV and another proactive session is ongoing, proactive commands in the script shall be silently ignored.

In case of "early response", the response to the sending entity shall be sent before processing the rest of the command TLVs. The number of executed commands TLV objects shall include all objects up to the immediate action TLV encoding the "early response". No other response data shall be sent after the response sent due to the early response action TLV.

NOTE: This is useful in case of some refresh modes, where the UICC might not be able to send a response after the refresh is performed by the terminal.

Proactive commands as defined in table 5.5 are allowed as Immediate Action. The behaviour of the card for other commands is undefined.

Table 5.5: Allowed proactive commands for Immediate Action

DISPLAY TEXT
PLAY TONE
REFRESH

5.2.1.3 Error Action TLV

The Error Action TLV is a BER-TLV data object that allows the Remote Management Application to issue a proactive command in case of error in the execution. It shall be formatted as shown in tables 5.6, 5.7 or 5.8.

The Error Action tag is defined in annex A.

Table 5.6: Error Action TLV - normal format

Length in bytes	Name
1	Error Action tag
L	Length of Error Action = A > 1
A	Set of COMPREHENSION-TLV data objects

Table 5.7: Error Action TLV - referenced format

Length in bytes	Name
1	Error Action tag
1	Length of Error Action = 1
1	'01' to '7F': Reference to a record in EF_{RMA} other values: RFU

Table 5.8: Error Action TLV - no action

Length in bytes	Name
1	Error Action tag
1	Length of Error Action = 0

In case of referenced format, the referenced record in EF_{RMA} shall contain the set of COMPREHENSION-TLV data objects preceded by a length value as defined for a BER-TLV, see ETSI TS 123 048 [10].

Proactive commands as defined in table 5.9 are allowed as Error Action. The behaviour of the card for other commands is undefined.

Table 5.9: Allowed proactive commands for Error Action

DISPLAY TEXT
PLAY TONE

If an error is encountered when processing a C-APDU, error actions shall be performed as follows:

- If there is an Error Action TLV between the start of the script and the C-APDU resulting in an error, the action defined in the last Error Action TLVs shall be performed. If this last Error Action TLV has zero length, no action shall be performed.
- If there is no Error Action TLV between the start of the script and the C-APDU resulting in an error, no action shall be performed.

5.2.1.4 Script Chaining TLV

The optional Script Chaining TLV is a BER-TLV data object and shall be coded as shown in table 5.9a.

Table 5.9a: Script Chaining TLV

Length in bytes	Name
1	Script Chaining tag
1	Script Chaining Length = 1
1	Script Chaining Value