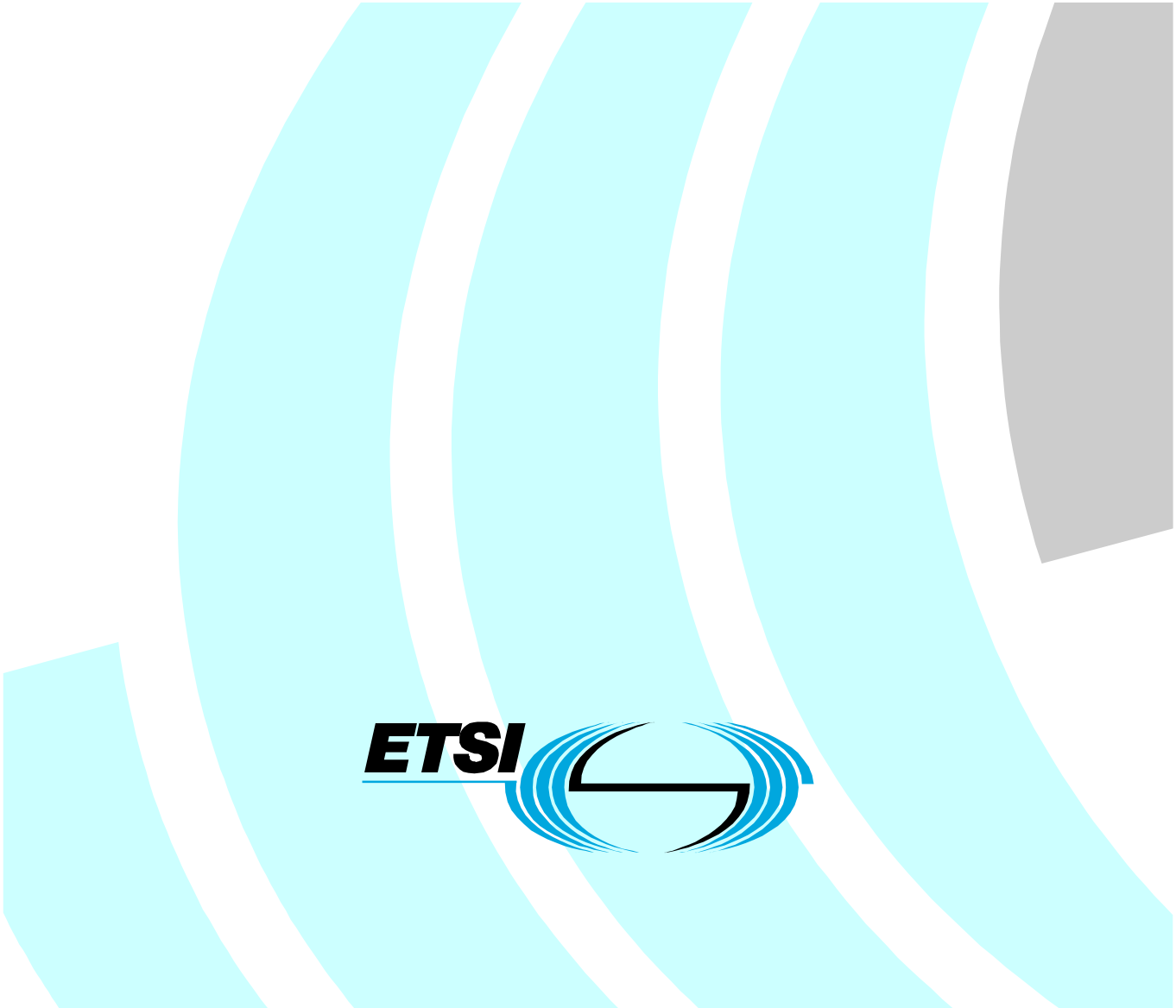
ETSI TS 102 225 V6.2.0 (2003-06)

Technical Specification

Smart cards;
Secured packet structure for UICC based applications
(Release 6)



Reference
RTS/SCP-000283r2

Keywords
smart card, security

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 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is 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 (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, 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.

Foreword

This Technical Specification (TS) has been produced by ETSI Project Smart Card Platform (SCP).

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 EP SCP and may change following formal EP SCP approval. If EP SCP 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 EP SCP for information;
  - 2 presented to EP SCP for approval;
  - 3 or greater indicates EP SCP 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.

1 Scope

The present document specifies the structure of the Secured Packets in a general format.

It is applicable to the exchange of secured packets between an entity in a network and an entity in the UICC.

Secured Packets contain application messages to which certain mechanisms according to TS 102 224 [1] have been applied. Application messages are commands or data exchanged between an application resident in or behind the network and on the UICC. The Sending/Receiving Entity in the network and the UICC are responsible for applying the security mechanisms to the application messages and thus turning them into Secured Packets.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

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

[1] ETSI TS 102 224: "Smart cards; Security mechanisms for UICC based Applications - Functional requirements (Release 6)".

[2] ISO/IEC 7816-6 (1998): "Identification cards - Integrated circuit(s) cards with contacts - Part 6: Interindustry data elements".

[3] ISO 8731-1 (1987): "Banking - Approved algorithms for message authentication - Part 1: DEA".

[4] ISO/IEC 10116 (1997): "Information technology - Security techniques - Modes of operation for an n-bit block cipher".

[5] ETSI TS 131 115: "Universal Mobile Telecommunications System (UMTS); Secured packet structure for (Universal) Subscriber Identity Module ((U)SIM) Toolkit applications (3GPP TS 31.115)".

[6] "Open Platform Card Specification version 2.0.1" (see http://www.globalplatform.org/).

[7] ISBN 0-471-12845-7 (1996): "Applied Cryptography: Protocols, Algorithms, and Source Code in C, 2nd Edition", Bruce Schneier, John Wiley & Sons.

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

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

application layer: layer above the Transport Layer on which the Application Messages are exchanged between the Sending and Receiving Applications

application message: package of commands or data sent from the Sending Application to the Receiving Application, or vice versa, independently of the transport mechanism

NOTE: An Application Message is transformed with respect to a chosen Transport Layer and chosen level of security into one or more secured packets.

card manager: application in charge of application management as defined in the Open Platform Card Specification (see "Open Platform Card Specification version 2.0.1").

command header: Security Header of a Command Packet. It includes all fields except the Secured Data.

command packet: Secured Packet transmitted by the Sending Entity to the Receiving Entity, containing a secured Application Message.

counter: mechanism or data field used for keeping track of a message sequence

NOTE: This could be realised as a sequence oriented or time stamp derived value, maintaining a level of synchronisation between the Sending Entity and the Receiving Entity.

cryptographic checksum: string of bits derived from some secret information, (e.g. a secret key), part or all of the Application Message, and possible further information (e.g. part of the Security Header)

NOTE: The secret key is known to the Sending Entity and to the Receiving Entity. The Cryptographic Checksum is often referred to as Message Authentication Code.

DES: standard cryptographic algorithm specified as DEA in ISO 8731-1

digital signature: string of bits derived from some secret information, (e.g. a secret key), the complete Application Message, and possible further information (e.g. part of the Security Header)

NOTE: The secret information is known only to the Sending Entity. Although the authenticity of the Digital Signature can be proved by the Receiving Entity, the Receiving Entity is not able to reproduce the Digital Signature without knowledge of the secret information owned by the Sending Entity.

receiving application: entity to which the Application Message is destined

receiving entity:entity where the Secured Packet is received (e.g. SMS-SC, UICC, USSD entry point, or dedicated Toolkit Server) and where the security mechanisms are utilised

NOTE: The Receiving Entity processes the Secured Packets.

redundancy check: string of bits derived from the Application Message and possible further information for the purpose of detecting accidental changes to the message, without the use of any secret information

response header: Security Header of a Response Packet

response packet: Secured Packet transmitted by the Receiving Entity to the Sending Entity, containing a secured response and possibly application data

secured data: this field contains the Secured Application Message and possibly padding octets

secured packet: information flow on top of which the level of required security has been applied

NOTE: An Application Message is transformed with respect to a chosen Transport Layer and chosen level of security into one or more Secured Packets.

security domain: application in charge of security management as defined in the Open Platform Card Specification (see "Open Platform Card Specification version 2.0.1")

security header: that part of the Secured Packet which consists of all security information (e.g. counter, key identification, indication of security level, checksum or Digital Signature)

sender identification: simple verification of the identity of the Sending Entity by the Receiving Entity comparing the sender identity with an apriori stored identity of the sender at the Receiving Entity

sending application: entity generating an Application Message to be sent

sending entity: entity from which the Secured Packet originates (e.g. SMS-SC, UICC, USSD entry point, or dedicated Toolkit Server) and where the security mechanisms are invoked. The Sending Entity generates the Secured Packets to be sent

status code:indication that a message has been received (correctly or incorrectly, indicating reason for failure)

transport layer: layer responsible for transporting Secured Packets through the network

NOTE: The transport layer implements one or more transport mechanisms, (e.g. SMS or USSD).

unsecured acknowledgement: Status Code included in a response message

3.2 Abbreviations

For the purpose of the present document, the following abbreviations apply:

CBC	Cipher Block Chaining
CC	Cryptographic Checksum
CHI	Command Header Identifier
CHL	Command Header Length
CNTR	Counter
CPI	Command Packet Identifier
CPL	Command Packet Length
DES	Data Encryption Standard
DS	Digital Signature
ECB	Electronic codebook
GP	GlobalPlatform
KIc	Key and algorithm Identifier for ciphering
KID	Key and algorithm Identifier for RC/CC/DS
NAA	Network Access Application
PCNTR	Padding Counter
PoR	Proof of Receipt
RA	Receiving Application
RC	Redundancy Check
RE	Receiving Entity
RHI	Response Header Identifier
RHL	Response Header Length
RPI	Response Packet Identifier
RPL	Response Packet Length
SA	Sending Application
SE	Sending Entity
SM	Short Message
SMS	Short Message Service
SMS-CB	Short Message Service – Cell Broadcast
SMS-SC	Short Message Service - Service Centre
SPI	Security Parameters Indication
TAR	Toolkit Application Reference
TLV	Tag – Length – Value (data structure)
USSD	Unstructured Supplementary Services Data

4 Overview of Security System

An overview of the secure communication related to the Card Application Toolkit together with the required security mechanisms is given in TS 102 224 [1] (see figure 1).

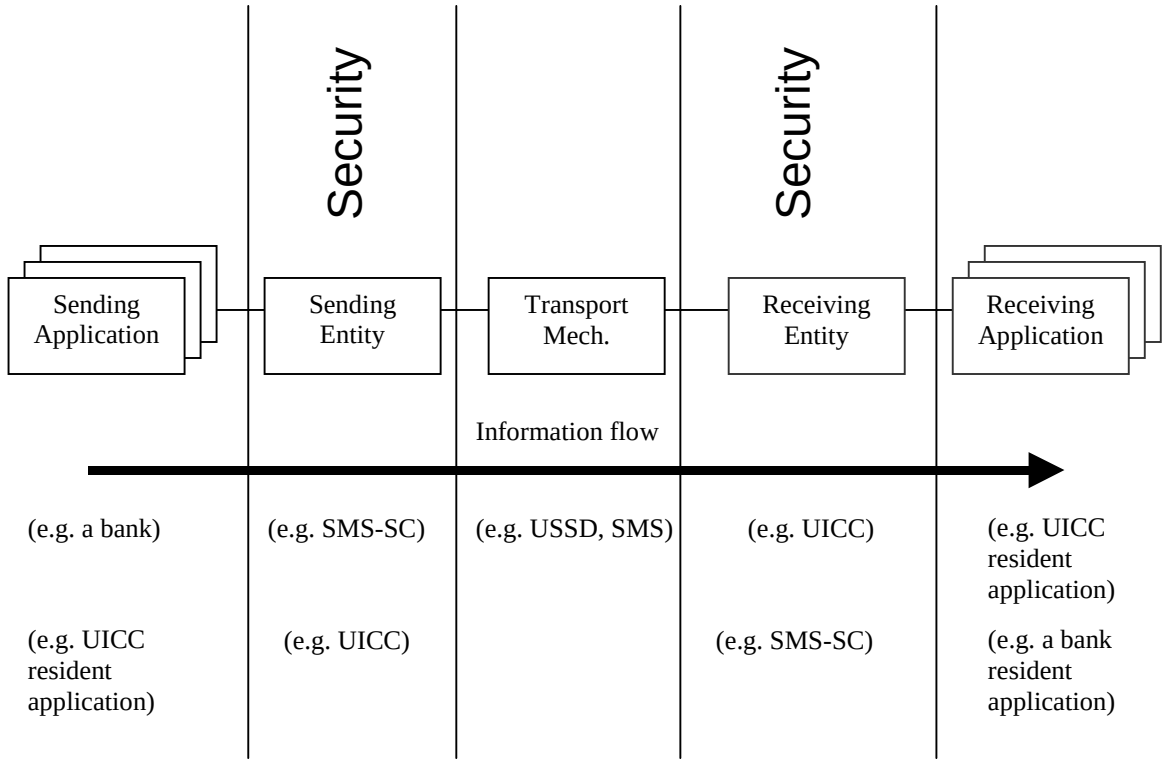


Figure 1: System overview

The Sending Application prepares an Application Message and forwards it to the Sending Entity, with an indication of the security to be applied to the message.

The Sending Entity prepends a Security Header (the Command Header) to the Application Message. It then applies the requested security to part of the Command Header and all of the Application Message, including any padding octets. The resulting structure is here referred to as the (Secured) Command Packet.

Under normal circumstances the Receiving Entity receives the Command Packet and unpacks it according to the security parameters indicated in the Command Header. The Receiving Entity subsequently forwards the Application Message to the Receiving Application indicating to the Receiving Application the security that was applied. The interface between the Sending Application and Sending Entity and the interface between the Receiving Entity and Receiving Application are proprietary and therefore outside the scope of the present document.

If so indicated in the Command Header, the Receiving Entity shall create a (Secured) Response Packet. The Response Packet consists of a Security Header (the Response Header) and optionally, application specific data supplied by the Receiving Application. Both the Response Header and the application specific data are secured using the security mechanisms indicated in the received Command Packet. The Response Packet will be returned to the Sending Entity, subject to constraints in the transport layer, (e.g. timing).

Although in some cases there might be no direct acknowledgement mechanism (i.e. for SMS-CB) the Sending Application may have requested a response. In this case a (Secured) Response Packet could be sent using a different bearer by the Receiving Application.

In some circumstances a security related error may be detected at the Receiving Entity. In such circumstances the Receiving Entity shall react according to the following rules:

1) nothing shall be forwarded to the Receiving Application. i.e. no part of the Application Message, and no indication of the error;

2) if the Sending Entity does not request a response (in the Command Header) the Receiving Entity discards the Command Packet and no further action is taken;

3) if the Sending Entity does request a response and the Receiving Entity can unambiguously determine what has caused the error, the Receiving Entity shall create a Response Packet indicating the error cause. This Response Packet shall be secured according to the security indicated in the received Command Packet;

4) if the Sending Entity does request a response and the Receiving Entity cannot determine what has caused the error, the Receiving Entity shall send a Response Packet indicating that an unidentified error has been detected. This Response Packet is sent without any security being applied;

5) if the Receiving Entity receives an unrecognisable Command Header (e.g. an inconsistency in the Command Header), the Command Packet shall be discarded and no further action taken.

5 Generalised Secured Packet structure

Command and Response Packets have the same overall structure consisting of a variable length security header within a variable length shell. To model this, use is made of a double TLV -tag, length, value- structure.

5.1 Command Packet structure

The Command Header precedes the Secured Data in the Command Packet, and is of variable length.

The Command Packet shall be structured according to table 1.

Table 1: Structure of the Command Packet

Element	Length	Comment
Command Packet Identifier (CPI)	1 octet	Identifies that this data block is the secured Command Packet.
Command Packet Length (CPL)	variable	This shall indicate the number of octets from and including the Command Header Identifier to the end of the Secured Data, including any padding octets required for ciphering.
Command Header Identifier (CHI)	1 octet	Identifies the Command Header.
Command Header Length (CHL)	variable	This shall indicate the number of octets from and including the SPI to the end of the RC/CC/DS.
Security Parameter Indicator (SPI)	2 octets	see detailed coding in clause 5.1.1.
Ciphering Key Identifier (KIc)	1 octet	Key and algorithm Identifier for ciphering.
Key Identifier (KID)	1 octet	Key and algorithm Identifier for RC/CC/DS.
Toolkit Application Reference (TAR)	3 octets	Coding is application dependent as defined in TS 101 220 [8].
Counter (CNTR)	5 octets	Replay detection and Sequence Integrity counter.
Padding counter (PCNTR)	1 octet	This indicates the number of padding octets used for ciphering at the end of the secured data.
Redundancy Check (RC), Cryptographic Checksum (CC) or Digital Signature (DS)	variable	Length depends on the algorithm. A typical value is 8 octets if used, and for a DS could be 48 or more octets; the minimum should be 4 octets.
Secured Data	variable	Contains the Secured Application Message and possibly padding octets used for ciphering.

Unless indicated otherwise, the CPL and the CHL shall be coded according to ISO/IEC 7816-6 [2].

Table 2: Linear Representation of Command Packet

CPI	CPL	CHI	CHL	SPI	KIc	KID	TAR	CNTR	PCNTR	RC/CC/DS	Secured Data with Padding
								note 1	note 1	note 1	note 1
	note 3		note 3	note 2	note 2	note 2	note 2	note 2	note 2		note 2

NOTE 1: These fields are included in the data to be ciphered if ciphering is indicated in the Security Header.
NOTE 2: These fields are included in the calculation of the RC/CC/DS.
NOTE 3: Part or all of these fields may also be included in the calculation of the RC/CC/DS, depending on implementation (e.g. SMS).

If ciphering is indicated, first the RC/CC/DS shall be calculated as indicated in note 2, and then ciphering shall be applied, as indicated in note 1.

If the SPI indicates that a specific field is unused, the Sending Entity shall set the contents of this field to zero, and the Receiving Entity shall ignore the contents.

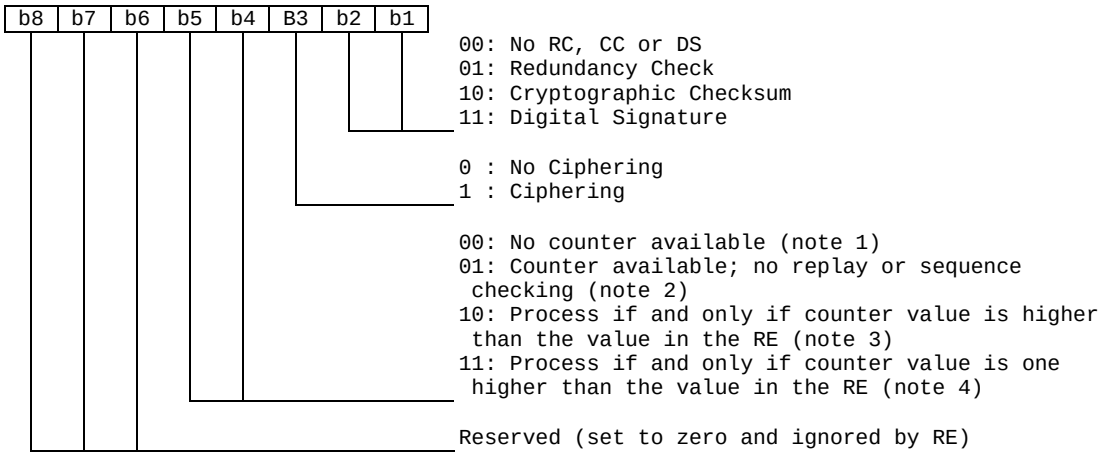
If the SPI indicates that no RC, CC or DS is present in the Command Header, the RC/CC/DS field shall be of zero length.

If the Padding Counter content is zero, this shall indicate no padding octets, or no padding is necessary.

5.1.1 Coding of the SPI

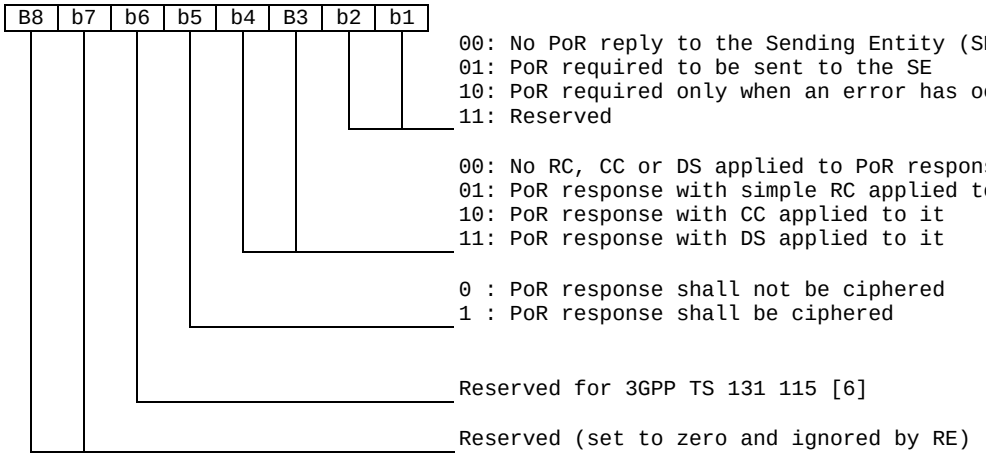
The SPI is coded as below.

First Octet:



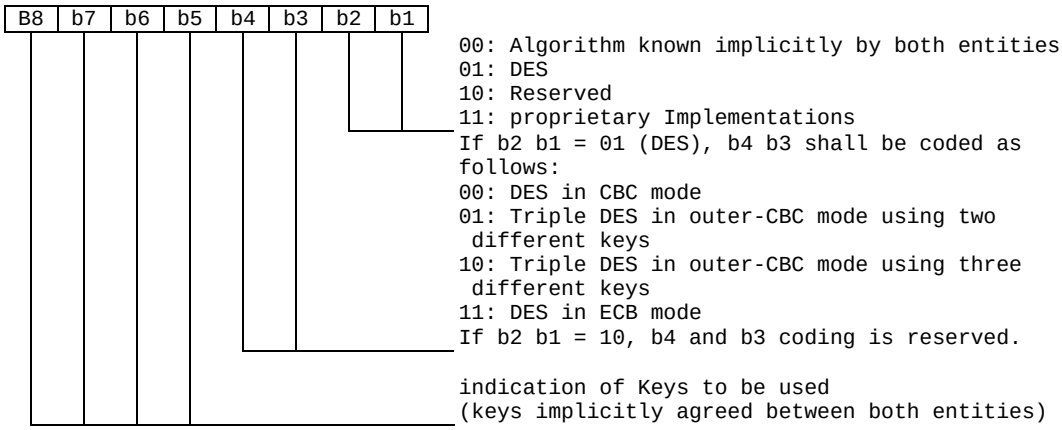
NOTE 1: In this case the counter field is present in the message.
NOTE 2: In this case the counter value is used for information purposes only, (e.g. date or time stamp). If the Command Packet was successfully unpacked, the counter value can be forwarded from the Receiving Entity to the Receiving Application. This depends on proprietary implementations and happens in an application dependent way.
NOTE 3: The counter value is compared with the counter value of the last received Command Packet. This is tolerant to failures on the transport level (i.e. losses of Command Packets). A possible scenario is a global update.
NOTE 4: This provides strict control in addition to security indicated in note 3.

Second Octet:



5.1.2 Coding of the KIc

The KIc is coded as below.



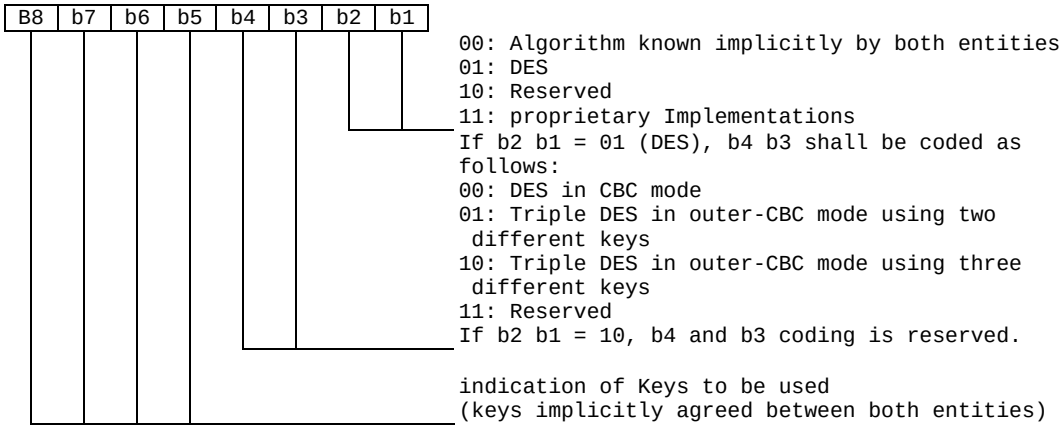
DES is the algorithm specified as DEA in ISO 8731-1 [3]. DES in CBC mode is described in ISO/IEC 10116 [4]. Triple DES in outer-CBC mode is described in clause 15.2 of [7]. DES in ECB mode is described in ISO/IEC 10116 [4].

The initial chaining value for CBC modes shall be zero.

For GlobalPlatform security architecture compliant cards see annex A.

5.1.3 Coding of the KID

The KID is coded as below.



DES is the algorithm specified as DEA in ISO 8731-1 [3]. DES in CBC mode is described in ISO/IEC 10116 [4]. Triple DES in outer-CBC mode is described in clause 15.2 of [20].

The initial chaining value for CBC modes shall be zero. If padding is required, the padding octets shall be coded hexadecimal '00'. These octets shall not be included in the secured data.

For GlobalPlatform security architecture compliant cards see annex A.

5.1.4 Counter Management

If in the first SPI byte b4b5=00 (No counter available) the counter field shall be ignored by the RE and the RE shall not update the counter.

If b5 of the first SPI byte is equal to 1 then the following rules shall apply to counter management, with the goal of preventing replay and synchronisation attacks:

- The SE sets the counter value. It shall only be incremented.
- The RE shall update the counter to its next value upon receipt of a Command Packet after the corresponding security checks (i.e. RC/CC/DS and CNTR verification) have been passed successfully.

  The next counter value is the one received in the incoming message.
- When the counter value reaches its maximum value the counter is blocked.

If there is more than one SE, care has to be taken to ensure that the counter values remain synchronized between the SE's to what the RE is expecting, irrespective of the transport mechanism employed.

The level of security is indicated via the proprietary interface between the Sending/Receiving Application and Sending/Receiving Entity. Application designers should be aware that if the Sending Application requests "No RC/CC/DS" or "Redundancy Check" and "No Counter Available" from the SE, no security is applied to the Application Message and therefore there is an increased threat of malicious attack.

For GlobalPlatform security architecture compliant cards see annex A.

5.2 Response Packet structure

Table 3: Structure of the Response Packet

Element	Length	Comment
Response Packet Identifier (RPI)	1 octet	Identifies a Response Packet.
Response Packet Length (RPL)	variable	Indicates the number of octets from and including RHI to the end of Additional Response data, including any padding octets.
Response Header Identifier (RHI)	1 octet	Identifies the Response Header.
Response Header Length (RHL)	variable	Indicates the number of octets from and including TAR to the end of the RC/CC/DS.
Toolkit Application Reference (TAR)	3 octets	This shall be a copy of the contents of the TAR in the Command Packet.
Counter (CNTR)	5 octets	This shall be a copy of the contents of the CNTR in the Command Packet.
Padding counter (PCNTR)	1 octet	This indicates the number of padding octets at the end of the Additional Response Data.
Response Status Code Octet	1 octet	Codings defined in table 5.
Redundancy Check (RC), Cryptographic Checksum (CC) or Digital Signature (DS)	variable	Length depending on the algorithm indicated in the Command Header in the incoming message. A typical value is 4 octets to 8 octets, or zero if no RC/CC/DS is requested.
Additional Response Data	variable	Optional Application Specific Response Data, including possible padding octets.

Unless indicated otherwise, the RPL and RHL shall be coded according to ISO/IEC 7816-6 [2].

Table 4: Linear Representation of Response Packet

RPI	RPL	RHI	RHL	TAR	CNTR	PCNTR	Status Code	RC/CC/DS	Additional Response Data with padding
					note 1	note 1	note 1	note 1	note 1
	note 3		note 3	note 2	note 2	note 2	note 2		note 2

NOTE 1: If ciphering is indicated in the Command Packet SPI then these fields shall be ciphered.
NOTE 2: These fields shall be included in the calculation of the RC/CC/DS.
NOTE 3: Part or all of these fields may also be included in the calculation of the RC/CC/DS, depending on implementation (e.g. SMS).

If ciphering is indicated, first the RC/CC/DS shall be calculated as indicated in note 2, and then ciphering shall be applied, as indicated in note 1.

If the SPI indicates that a specific field is unused, than its contents shall be set to zero, and ignored by the recipient of the Response Packet.

If the SPI in the Command Packet indicates that no RC, CC or DS is present in the Command Header, this field shall be of zero length.

If the Padding Counter content is zero, this shall indicate no padding octets are present, or no padding is necessary.

Table 5: Response Status Codes

Status Code (hexadecimal)	Meaning
'00'	PoR OK.
'01'	RC/CC/DS failed.
'02'	CNTR low.
'03'	CNTR high.
'04'	CNTR Blocked
'05'	Ciphering error.
'06'	Unidentified security error. This code is for the case where the Receiving Entity cannot correctly interpret the Command Header and the Response Packet is sent unciphered with no RC/CC/DS.
'07'	Insufficient memory to process incoming message.
'08'	This status code "more time" should be used if the Receiving Entity/Application needs more time to process the Command Packet due to timing constraints. In this case a later Response Packet should be returned to the Sending Entity once processing has been completed.
'09'	TAR Unknown
'0A'	Insufficient security level
'0B' - 'FF'	Reserved for future use.

Annex A (normative): Relation between security layer and GlobalPlatform security architecture

This annex only applies to cards implementing the security architecture defined in the Open Platform Card Specification [6].

The security of Application Messages (i.e. RC/CC/DS, ciphering/deciphering, counter management) shall be managed by the Security Domain of the Application.

A.1 Key set version - counter association within a Security Domain

A separate and different counter shall be associated to each key set version as described in table A.1.

Table A.1

	Key Set Version 0	**Key Set Version 1**	**….**	**Key Set Version n (maximum 'F')**
	Reserved	**Counter 1**		**Counter n**
Key Index 1	Reserved	KIc 1		KIc n
Key Index 2	Reserved	KID 1		KID n
Key Index 3	Reserved	KIK 1		KIK n

A.2 Security keys KIc, KID

The indication of the key to be used in the KIc and KID fields shall refer to an GlobalPlatform key set version number.

The algorithm to be used with the key shall be the algorithm associated with the key (as described in the Open Platform Card specification [6]).

The key set version number indicated in the KIc and KID fields shall be identical when different from 0. If the key set version numbers are different (and both different from 0) then the message shall be rejected with the "Unidentified security error" Response Status Code.

Annex B (informative): Change History

This annex lists all changes made to the present document.

History Table								
Date	**Meeting**	**Tdoc**	**CR**	**Rv**	**Cat**	**Changes**	**Old**	**New**
2001-10	T3 API #9	T3a010195				Initial version is based on 3GPP TS 23.038 v5.1.0.	-	0.0.0
2001-11	T3#21/ SCP#8	T3-010669/ SCP-010345				Submitted to 3GPP T3#21 - EP SCP#8. Editorial changes.	0.0.0	0.0.1
2001-11	SCP#8	SCP-010376				Editorial and formatting enhancements. Version number raised to 1.0.0 in line with decision at EP SCP #8	0.0.1	1.0.0
2002-01	SCP2#2	SCP2-020018				Updated to include the results of TSG-T#14 and editorial changes.	1.0.0	1.1.0
2002-03	SCP#9	SCP-020048				Updated to include the results of TSG-T#15. Submitted to SCP#9 for approval.	1.1.0	2.0.0
2002-03	SCP#9	SCP-020056				Editorial changes after discussion at SCP#9. This version has been sent to the ETSI secretariat for publication in March 2002 as TS 102 225 v6.0.0. No technical changes compared to v2.0.0	2.0.0	6.0.0
2003-01	SCP#12	SCP-030021	001		A	Clarification on the RC/CC/DS coding in SPI2	6.0.0	6.1.0
2003-05	SCP#13	SCP-030166	002		F	Clarification of the description/usage of b3 b4 of KIc and KID	6.1.0	6.2.0
		SCP-030123	003		F	TAR coding clarification		

History

Document history		
V6.0.0	April 2002	Publication
V6.1.0	February 2003	Publication
V6.2.0	June 2003	Publication