GlobalPlatform Technology

Secure Channel Protocol '03'

Card Specification v2.3 – Amendment D

Version 1.1.2

Public Release

March 2019

Document Reference: GPC_SPE_014

THIS SPECIFICATION OR OTHER WORK PRODUCT IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NON-INFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY IMPLEMENTATION OF THIS SPECIFICATION OR OTHER WORK PRODUCT SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER THE COMPANY, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER DIRECTLY OR INDIRECTLY ARISING FROM THE IMPLEMENTATION OF THIS SPECIFICATION OR OTHER WORK PRODUCT.

Contents

1 Introduction 7
1.1 Audience 7
1.2 IPR Disclaimer 7
1.3 References 7
1.4 Terminology and Definitions 8
1.5 Abbreviations and Notations 8

2 Revision History 10

3 Use Cases and Requirements 12

4 Specification Amendments 14
4.1 Algorithm 14
4.1.1 Advanced Encryption Standard (AES) 14
4.1.2 Encryption/Decryption 14
4.1.3 MACing 14
4.1.4 AES Padding 14
4.1.5 Data Derivation Scheme 15

5 Secure Channel Protocol Usage 16
5.1 Secure Communication Configuration 16
5.2 Mutual Authentication 17
5.3 Message Integrity 19
5.4 Message Data Confidentiality 19
5.5 API and Security Level 20
5.6 Protocol Rules 21

6 Cryptographic Keys 22
6.1 AES Keys 22
6.2 Cryptographic Usage 23
6.2.1 AES Session Keys 23
6.2.2 Challenges and Authentication Cryptograms 24
6.2.2.1 Card Challenge 24
6.2.2.2 Card Authentication Cryptogram 24
6.2.2.3 Host Authentication Cryptogram 24
6.2.3 Message Integrity Using Explicit Secure Channel Initiation 24
6.2.4 APDU Command C-MAC Generation and Verification 25
6.2.5 APDU Response R-MAC Generation and Verification 26
6.2.6 APDU Command C-MAC and C-DECRYPTION Generation and Verification 29
6.2.7 APDU Response R-MAC and R-ENCRYPTION Generation and Verification 30
6.2.8 Key Sensitive Data Encryption and Decryption 31

7 Commands 32
7.1 Secure Channel Commands 33
7.1.1 INITIALIZE UPDATE Command 33
7.1.1.1 Definition and Scope 33
7.1.1.2 Command Message 33
7.1.1.3 Reference Control Parameter P1 – Key Version Number 33
7.1.1.4 Reference Control Parameter P2 – Key Identifier 33
7.1.1.5 Data Field Sent in the Command Message 34
7.1.1.6 Data Field Returned in the Response Message 34
7.1.1.7 Processing State Returned in the Response Message 34

7.1.2 EXTERNAL AUTHENTICATE Command....35
7.1.2.1 Definition and Scope....35
7.1.2.2 Command Message....35
7.1.2.3 Reference Control Parameter P1 – Security Level....35
7.1.2.4 Reference Control Parameter P2....36
7.1.2.5 Data Field Sent in the Command Message....36
7.1.2.6 Data Field Returned in the Response Message....36
7.1.2.7 Processing State Returned in the Response Message....36
7.1.3 BEGIN R-MAC SESSION Command....37
7.1.3.1 Definition and Scope....37
7.1.3.2 Command Message....37
7.1.3.3 Reference Control Parameter P1....37
7.1.3.4 Reference Control Parameter P2....38
7.1.3.5 Data Field Sent in the Command Message....38
7.1.3.6 Data Field Returned in the Response Message....38
7.1.3.7 Processing State Returned in the Response Message....38
7.1.4 END R-MAC SESSION Command....39
7.1.4.1 Definition and Scope....39
7.1.4.2 Command Message....39
7.1.4.3 Reference Control Parameter P1....39
7.1.4.4 Reference Control Parameter P2....39
7.1.4.5 Data Field Sent in the Command Message....39
7.1.4.6 Data Field Returned in the Response Message....39
7.1.4.7 Processing State Returned in the Response Message....40
7.2 PUT KEY Command (AES Key-DEK)....40
7.3 STORE DATA (AES Key-DEK)....40
8 AES for Card Content Management....41
8.1 DAPs for AES....41
8.2 Tokens for AES....41
8.3 Receipts for AES....41

Figures

Figure 5-1: Explicit Secure Channel Initiation Flow...18
Figure 6-1: APDU C-MAC Generation...25
Figure 6-2: APDU R-MAC Generation...26
Figure 6-3: MAC Chaining...27
Figure 6-4: APDU Command Data Field Encryption...29
Figure 6-5: APDU Response Data Field Encryption...30
Figure 6-6: Sensitive Data Encryption...31

Tables

Table 1-1: Normative References....7

Table 1-2: Abbreviations and Notations 8

Table 2-1: Revision History 10

Table 4-1: Data Derivation Constants 15

Table 5-1: Values of Parameter “i” 16

Table 6-1: Security Domain Secure Channel Keys....22

Table 6-2: AES Key Derivation Elements....23

Table 7-1: SCP03 Command Support....32

Table 7-2: INITIALIZE UPDATE Command Message....33

Table 7-3: INITIALIZE UPDATE Response Message 34

Table 7-4: INITIALIZE UPDATE Error Condition....34

Table 7-5: EXTERNAL AUTHENTICATE Command Message 35

Table 7-6: EXTERNAL AUTHENTICATE Reference Control Parameter P1....35

Table 7-7: EXTERNAL AUTHENTICATE Error Condition 36

Table 7-8: BEGIN R-MAC SESSION Command Message....37

Table 7-9: BEGIN R-MAC SESSION Reference Control Parameter P1....37

Table 7-10: BEGIN R-MAC SESSION Reference Control Parameter P2....38

Table 7-11: END R-MAC SESSION Command Message 39

Table 7-12: END R-MAC SESSION Reference Control Parameter P2 39

1 Introduction

This document proposes a new secure channel protocol based on AES keys and specifies:

- A new mechanism to generate session keys.
- The schemes to be used with AES for C-MAC, R-MAC, command data field encryption and response data field encryption.
- The format of PUT KEY for AES.

This new protocol is based on existing SCP01 and SCP02 protocols. It supports AES-based cryptography in lieu of TDEA. The protocol protects bidirectional communication between the Host and the card (decryption/MAC verification for incoming commands, encryption/MAC generation on card response).

In addition, the document defines the formats and requirements for DAPs, Tokens and Receipts if AES is used for card content management activities.

1.1 Audience

This amendment is intended primarily for card implementers, application developers, and off-card entities communicating with secure channel protocols.

It is assumed that the reader is familiar with smart cards and smart card production, and in particular familiar with the GlobalPlatform Card Specification ([GPCS]).

1.2 IPR Disclaimer

Attention is drawn to the possibility that some of the elements of this GlobalPlatform specification or other work product may be the subject of intellectual property rights (IPR) held by GlobalPlatform members or others. For additional information regarding any such IPR that have been brought to the attention of GlobalPlatform, please visit https://globalplatform.org/specifications/ip-disclaimers/. GlobalPlatform shall not be held responsible for identifying any or all such IPR, and takes no position concerning the possible existence or the evidence, validity, or scope of any such IPR.

1.3 References

Table 1-1: Normative References

Standard / Specification	Description	Ref
GlobalPlatform Card Specification	GlobalPlatform Card Specification v2.3	[GPCS]
ISO/IEC 8825-1	Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)	[ISO 8825-1]
NIST SP 800-108	Recommendation for Key Derivation Using Pseudorandom Functions, November 2008	[NIST 800-108]
NIST SP 800-38A	Recommendation for Block Cipher Modes of Operation: Methods and Techniques, 2001	[NIST 800-38A]

Standard / Specification	Description	Ref
NIST SP 800-38B	Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, May 2005	[NIST 800-38B]
NIST SP 800-57 Part 1 revised	Recommendation for Key Management – Part 1: General (Revised) March, 2007	[NIST 800-57]
NIST SP 800-78-1	Cryptographic Algorithms and Key Sizes for Personal Identity Verification, August 2007	[NIST 800-78-1]

1.4 Terminology and Definitions

Terms used in this document are defined in GlobalPlatform Card Specification ([GPCS]).

1.5 Abbreviations and Notations

Selected abbreviations and notations used in this document are included in Table 1-2. Additional abbreviations and notations are defined in [GPCS].

Table 1-2: Abbreviations and Notations

Abbreviation / Notation	Meaning
2TDEA	Two key Triple DEA
3TDEA	Three key Triple DEA
AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
C-DECRYPTION	Command Decryption
CMAC	Cipher-based MAC (see note)
C-MAC	Command MAC (see note)
DAP	Data Authentication Pattern
DEA	Data Encryption Algorithm
DES	Data Encryption Standard
ECC	Elliptic Curve Cryptography
ICV	Initial Chaining Vector
ISO	International Organization for Standardization
KDF	Key Derivation Function
Key-DEK	Data Encryption Key
Lc	Exact length of command data in a case 3 or case 4 command
Le	Maximum length of data expected in response to a case 2 or case 4 command
LV	Length Value

Abbreviation / Notation	Meaning
MAC	Message Authentication Code
PRF	Pseudorandom Function
R-ENCRYPTION	Response Encryption
R-MAC	Response MAC
RSA	Rivest / Shamir / Adleman asymmetric algorithm
SCP	Secure Channel Protocol
S-ENC	Secure Channel command and response encryption key
S-MAC	Secure Channel C-MAC session key
S-RMAC	Secure Channel R-MAC session key
TDEA	Triple DEA

Note: C-MAC is the abbreviation used in [GPCS] for the MAC appended to command APDUs. This is not to be confused with CMAC, which is the abbreviation for a MAC calculation scheme specified in NIST SP 800-38B ([NIST 800-38B]).

2 Revision History

GlobalPlatform technical documents numbered *n*.0 are major releases. Those numbered *n*.1, *n*.2, etc., are minor releases where changes typically introduce supplementary items that do not impact backward compatibility or interoperability of the specifications. Those numbered *n.n*.1, *n.n*.2, etc., are maintenance releases that incorporate errata and precisions; all non-trivial changes are indicated, often with revision marks.

Table 2-1: Revision History

Date	Version	Description
April 2009	1.0	Initial Public Release
September 2009	1.1	Public Release • Added Chapter 8: AES for Card Content Management.
July 2014	1.1.1	Public Release • Encrypting AES keys with an AES Key-DEK of same or higher strength is now only a recommendation. This constraint has been released to allow loading AES keys for DAP or Delegated Management through a DES-based Secure Channel session (i.e. SCP02), and to allow for more practical DES-to-AES migration strategies (including SCP02 to SCP03 migration). This recommendation is moved from section 7.2 to section 6.1 so that it applies both to the PUT KEY and the STORE DATA command. • Precision regarding the response status words for which no R-MAC shall be returned (section 6.2.5). • Precision for generation and verification of C-DECRYPTION (see section 6.2.6) and R-ENCRYPTION (see section 6.2.7). • Precision regarding the response to the BEGIN R-MAC Session command when the Current Security Level already indicates R-MAC (i.e. previously requested by the EXTERNAL AUTHENTICATE command) (section 7.1.3.6). • Precision regarding the effect of the END R-MAC Session command in case of processing error (section 7.1.4.1).

Date	Version	Description
March 2019	1.1.2	Public Release • SCP01, previously deprecated, has been removed from [GPCS], so selected material from [GPCS] former Appendix D has been moved to this specification: o Description of three levels of security in section 5.1. o Description of mutual authentication in section 5.2. o Specification of INITIALIZE UPDATE command in section 7.1.1. o Specification of EXTERNAL AUTHENTICATE command in section 7.1.2. • SCP02 has been deprecated and will be removed from [GPCS], so selected material from [GPCS] Appendix E has been moved to this specification: o Discussion of Explicit Secure Channel Initiation Flow, including Figure 5-1 in section 5.2. • Most of the content of Chapter 4, Specification Amendments, has been moved to [GPCS] section B.2. (Section 4.1.5, Data Derivation Scheme, remains in this document.) • Section 7.1 clarifies the content of P2 in the INITIALIZE UPDATE command message. • The content of section 7.2, PUT KEY Command (AES Key-DEK) has been moved to [GPCS] section 11.8. • The content of section 7.3, STORE DATA (AES Key-DEK) has been moved to [GPCS] section 11.11. • The content of Chapter 8, AES for Card Content Management, has been moved to [GPCS] Appendix C. Selected revisions are boxed in red rather than revision marked.

3 Use Cases and Requirements

This document proposes a specification addendum to support the following requirements:

The Secure Channel is used to personalize cards at Issuance and during Post-Issuance. The mode of the Secure Channel Protocol which uses pseudo-random card challenges allows the offline preparation of personalization scripts while the card is not present and the processing of these scripts on the card without an online connection to the entity that prepared the scripts.

When the personalization involves the loading of a cryptographic key, the transport key that secures the transmission must be at least as strong as the key being transmitted.

To assist in the determination of suitable transport keys, the US National Institute of Standards and Technology (NIST) has published a document called "Recommendation for Key Management". This document, freely available on the NIST website under the reference NIST 800-57 Part 1 ([NIST 800-57]), is a mandatory standard for federal use in the United States of America, and is also endorsed by many other governments around the world. ~~It is referred to by the more widely known FIPS 201, Personal Identity Verification ([FIPS 201-1]).~~ [NIST 800-57] provides the cryptographic strength of key based on its algorithm and its size.

It results that a 2TDEA key can be used as a transport key to encrypt another 2TDEA key, an RSA 1024 key, or an ECC key with f = 160-223, but cannot be used to encrypt the following keys:

- 3TDEA Length Keys
- RSA above 1024
- AES-128
- AES-192
- AES-256
- ECC (f = 224 and above)

Furthermore, a 3TDEA key used as a transport key can only encrypt another 3TDEA key, an RSA 2048, or an ECC key with f = 224-255, but cannot be used to encrypt:

- RSA above 2048
- AES-128
- AES-192
- AES-256
- ECC (f = 256 and above)

Since the above types of keys are starting to become available in the latest generation of Java Cards™, it becomes important that GlobalPlatform provides a mechanism by which such key could be loaded into the card.

NIST has also published another standard called "Cryptographic Algorithms and Key Sizes for Personal Identify Verification". This document, also freely available on the NIST website under the reference NIST SP 800-78-1 ([NIST 800-78-1]), is a mandatory standard for the Personal Identity Verification (PIV) cards for all US Federal employees and contractors. ~~It is referred to by the more widely known [FIPS 201-1].~~ According to this standard, the time period to use RSA 1024 for digital signature ~~expires~~ expired on the 31st of December 2008, and the 2TDEA Keys cannot be used for card authentication ~~after~~ since the 31st of December 2010.

According to the above standards, an AES key is more suitable for a transport key as:

- An AES-128 key can be used to encrypt 3TDEA Keys, RSA up to 3072, AES-128 and ECC with f up to 383.
- An AES-192 key can be used in addition to encrypt RSA up to 7680, AES-192 and ECC with f = 384-511.
- An AES-256 key can be used in addition to encrypt RSA up to 15360, AES-256 and ECC with f = 512+.

This should ensure the permanence of a secure channel based on AES from a crypto analysis standpoint for several years.

4 Specification Amendments

4.1 Algorithm

A new section B.2 has been created in [GPCS] to describe Advanced Encryption Standard (AES) as used in GlobalPlatform Card Specifications.

The content of the following sections has been moved to [GPCS]. The revision-marked deletions are omitted for readability.

4.1.1 Advanced Encryption Standard (AES)

The content of this section has been moved to [GPCS] section B.2.

4.1.2 Encryption/Decryption

The content of this section has been moved to [GPCS] section B.2.1.

4.1.3 MACing

The content of this section has been moved to [GPCS] section B.2.2.

4.1.4 AES Padding

The content of this section has been moved to [GPCS] section B.2.3.

4.1.5 Data Derivation Scheme

The following data derivation scheme is used to generate keys, pseudo-random card challenges or cryptograms:

Data derivation shall use KDF in counter mode as specified in NIST SP 800-108 ([NIST 800-108]). The PRF used in the KDF shall be CMAC as specified in [NIST 800-38B], used with full 16-byte output length.

The “fixed input data” plus iteration counter shall be the concatenation of the following items in the given sequence (note that [NIST 800-108] allows the reordering of input data fields as long as the order, coding and length of each field is unambiguously defined):

- A 12-byte “label” consisting of 11 bytes with value '00' followed by a 1-byte derivation constant as defined below.
- A 1-byte “separation indicator” with value '00'.
- A 2-byte integer “L” specifying the length in bits of the derived data (value '0040', '0080', '00C0', or '0100').
- A 1-byte counter “i” as specified in the KDF (which may take the values '01' or '02'; value '02' is used when “L” takes the values '00C0' and '0100', i.e. when the PRF of the KDF is to be called twice to generate enough derived data).
- The “context” parameter of the KDF. Its content is further specified in the sections below applying the data derivation scheme.

Definition of the derivation constant:

Table 4-1: Data Derivation Constants

b8	b7	b6	b5	b4	b3	b2	b1	Description
0	0	0	0	0	0	0	x	authentication cryptogram generation
0	0	0	0	0	0	0	0	- card cryptogram
0	0	0	0	0	0	0	1	- host cryptogram
0	0	0	0	0	0	1	0	card challenge generation
0	0	0	0	0	1	x	x	key derivation
0	0	0	0	0	1	0	0	- derivation of S-ENC
0	0	0	0	0	1	1	0	- derivation of S-MAC
0	0	0	0	0	1	1	1	- derivation of S-RMAC
all other values								RFU

5 Secure Channel Protocol Usage

This chapter defines the usage of Secure Channel Protocol '03'.

5.1 Secure Communication Configuration

The three levels of security ~~are~~ supported ~~as defined in [GPCS] section D.1.1~~ by SCP03 are:

- Mutual authentication – The card and the off-card entity each prove that they have knowledge of the same secrets.
- Integrity and data origin authentication – The card ensures that the data being received from the off-card entity actually came from an authenticated off-card entity in the correct sequence and has not been altered.
- Confidentiality – Data being transmitted from the off-card entity to the card is not viewable by an unauthorized entity.

In SCP03, the “i” parameter is formed as a bit map on one byte as follows:

Table 5-1: Values of Parameter “i”

b8	b7	b6	b5	b4	b3	b2	b1	Description
				X	X	X	X	RFU (set to 0)
			0					Random card challenge
			1					Pseudo-random card challenge
	0	0						No R-MAC/R-ENCRYPTION support
	0	1						R-MAC support / no R-ENCRYPTION support
	1	1						R-MAC and R-ENCRYPTION support
X								Reserved

Note: “i” is a sub identifier within an object identifier, and bit b8 is reserved for use in the structure of the object identifier according to ISO/IEC 8825-1 ([ISO 8825-1]).

5.2 Mutual Authentication

Mutual authentication is achieved through the process of initiating a Secure Channel and provides assurance to both the card and the off-card entity that they are communicating with an authenticated entity. If any step in the mutual authentication process fails, the process shall be restarted, i.e. new challenges and Secure Channel Session keys shall be generated.

Initiating a Secure Channel was previously defined in [GPCS] Appendix D and referenced from this section.

The Secure Channel is explicitly initiated by the off-card entity using the INITIALIZE UPDATE and EXTERNAL AUTHENTICATE commands. The application may pass the APDU to the Security Domain using the appropriate API; e.g. the processSecurity() method of a GlobalPlatform Java Card.

The explicit Secure Channel initiation allows the off-card entity to instruct the card (see section 7.1.2, EXTERNAL AUTHENTICATE Command) as to the level of security required for the current Secure Channel (integrity and/or confidentiality) and to apply this level of security to all subsequent messages exchanged between the card and the off-card entity until the end of the Secure Channel Session. It also gives the off-card entity the possibility of selecting the Key Version Number and Key Identifier to be used (see section 7.1.1, INITIALIZE UPDATE Command).

Note: The explicit Secure Channel initiation also allows the card to inform the off-card entity what Secure Channel Protocol is supported, using the returned Secure Channel Protocol identifier.

The Secure Channel is always initiated (see section 7.1.1, INITIALIZE UPDATE Command) by the off-card entity by passing a ‘host’ challenge (random data unique to this Secure Channel Session) to the card.

The card, on receipt of this challenge, generates its own ‘card’ challenge (again random data unique to this Secure Channel Session).

The card, using the host challenge, the card challenge, and its internal static keys, creates new secret Secure Channel session keys and generates a first cryptographic value (card cryptogram) using one of its newly created Secure Channel session keys (see section 6.2.1).

This card cryptogram, the card challenge, the Secure Channel Protocol identifier, and other data is transmitted back to the off-card entity.

As the off-card entity should now have all the same information that the card used to generate the card cryptogram, it should be able to generate the same Secure Channel session keys and the same card cryptogram and by performing a comparison, it is able to authenticate the card.

The off-card entity now uses a similar process to create a second cryptographic value (host cryptogram) to be passed back to the card (see section 7.1.2, EXTERNAL AUTHENTICATE Command).

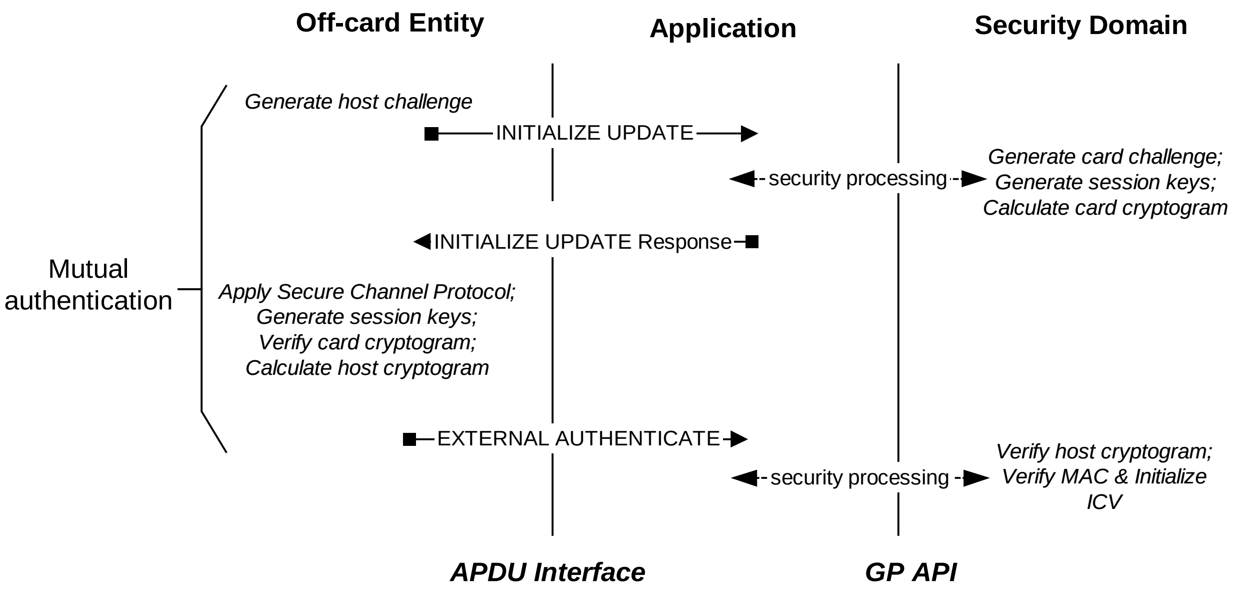
As the card has all the same information that the host used to generate the host cryptogram, it should be able to generate the same host cryptogram and, by performing a comparison, it is able to authenticate the off-card entity.

The mutual authentication flow was previously defined in [GPCS] Appendix E and referenced from this section.

Explicit Secure Channel Initiation Flow

The following flow is an example of explicit Secure Channel initiation between a card and an off-card entity. Expanding the authentication process shown in the flow described in [GPCS] section 7.3.1, Security Domain Support for Secure Messaging, it can be seen how an application would use the services of a Security Domain to achieve the explicit Secure Channel initiation.

Figure 5-1: Explicit Secure Channel Initiation Flow



5.3 Message Integrity

The C-MAC is generated by applying the NIST CMAC calculation (using S-MAC session key generated during the mutual authentication process) across the header and data field of an APDU command.

The card, on receipt of the message containing a C-MAC, using the same Secure Channel session key, performs the same operation and by comparing its internally generated C-MAC with the C-MAC received from the off-card entity is assured of the integrity of the full command.

If message data confidentiality has also been applied to the message, the C-MAC applies to the message data field after encryption has been performed.

The integrity of the sequence of commands being transmitted to the card is achieved by using the 16-byte C-MAC of a command as part of the input for the computation of the C-MAC of the next command. At any point in time, the last 16-byte C-MAC computed is part of the channel state and is referred to as the “MAC chaining value” further in this document. The first “MAC chaining value” is set to 16 bytes '00' (see section 6.2.3). This chaining (see Figure 6-3) ensures the card that all commands in a sequence have been received.

The integrity of the response is chained to the command sequence integrity by using the “MAC chaining value” as input as well for the computation of the R-MAC on responses (see Figure 6-3).

5.4 Message Data Confidentiality

The message data field is encrypted as specified in ~~section 4.1.2~~ [GPCS] section B.2.1 (using S-ENC Channel session key generated during the mutual authentication process) across the entire data field of the command message to be transmitted to the card, and if required also across the response transmitted from the card, regardless of its contents (clear text data and/or already protected sensitive data).

5.5 API and Security Level

A card implementing SCP03 shall implement the `SecureChannel` interface of the API specified in GlobalPlatform Card Specification ([GPCS]).

The following shall apply for the Security Level:

The Current Security Level of a communication not included in a Secure Channel Session shall be set to NO_SECURITY_LEVEL.

For Secure Channel Protocol '03', the Current Security Level established in a Secure Channel Session is a bitmap combination of the following values: AUTHENTICATED, C_MAC, R_MAC, C_DECRYPTION and R_ENCRYPTION.

The Current Security Level shall be set as follows:

- NO_SECURITY_LEVEL when a Secure Channel Session is terminated or not yet fully initiated;
- AUTHENTICATED after successful processing of an EXTERNAL AUTHENTICATE command: AUTHENTICATED shall be cleared once the Secure Channel Session is terminated;
- C_MAC after successful processing of an EXTERNAL AUTHENTICATE command with P1 indicating C-MAC (P1 = 'x1' or 'x3'): C_MAC shall be cleared once the Secure Channel Session is terminated. Note that C_MAC is always combined with AUTHENTICATED and simultaneously set and cleared;
- C_DECRYPTION after successful processing of an EXTERNAL AUTHENTICATE command with P1 indicating Command Encryption (P1= 'x3'): C_DECRYPTION shall be cleared once the Secure Channel Session is terminated. Note that C_DECRYPTION is always combined with AUTHENTICATED and C_MAC and simultaneously set and cleared;
- R_MAC after successful processing of an EXTERNAL AUTHENTICATE command with P1 indicating R-MAC (P1='1x' or '3x'): R_MAC shall be cleared once the Secure Channel Session is terminated. Note that in this case R_MAC is always combined with AUTHENTICATED and simultaneously set and cleared. R_MAC may also be combined with C_MAC or C_DECRYPTION (according to the P1 value of the EXTERNAL AUTHENTICATE command) and simultaneously set and cleared;
- R_ENCRYPTION after successful processing of an EXTERNAL AUTHENTICATE command with P1 indicating Response Encryption (P1= '3x'): R_ENCRYPTION shall be cleared once the Secure Channel Session is terminated. Note that R_ENCRYPTION is always combined with AUTHENTICATED and R_MAC and simultaneously set and cleared;
- R_MAC and no R_ENCRYPTION after successful processing of a BEGIN R-MAC SESSION command: R_MAC shall be cleared after successful processing of an END R-MAC SESSION command. Note that in this case R_MAC is combined with AUTHENTICATED and C_MAC or AUTHENTICATED, C_MAC and C_DECRYPTION depending on the pre-existing Current Security Level of the Secure Channel Session. R_MAC is set and cleared independently of AUTHENTICATED, C_MAC or C_DECRYPTION.
- R_MAC and R_ENCRYPTION after successful processing of a BEGIN R-MAC SESSION command: R_MAC and R_ENCRYPTION shall be cleared after successful processing of an END R-MAC SESSION command. Note that in this case R_MAC and R_ENCRYPTION are combined with AUTHENTICATED, C_MAC and C_DECRYPTION. R_MAC and R_ENCRYPTION are set and cleared independently of AUTHENTICATED, C_MAC or C_DECRYPTION.

5.6 Protocol Rules

In accordance with the general rules described in [GPCS] Chapter 10, the following protocol rules apply to Secure Channel Protocol '03':

- The successful initiation of a Secure Channel Session shall set the Current Security Level to the security level indicated in the EXTERNAL AUTHENTICATE command: it is at least set to AUTHENTICATED.
- The Current Security Level shall apply to the entire Secure Channel Session unless successfully modified at the request of the Application.
- When the Current Security Level is set to NO_SECURITY_LEVEL:
    - If the Secure Channel Session was aborted during the same Application Session, the incoming command shall be rejected with a security error;
    - Otherwise no security verification of the incoming command shall be performed. The Application processing the command is responsible to apply its own security rules.
- If a Secure Channel Session is active (i.e. Current Security Level at least set to AUTHENTICATED), the security of the incoming command shall be checked according to the Current Security Level regardless of the command secure messaging indicator:
    - When the security of the command does not match (nor exceeds) the Current Security Level, the command shall be rejected with a security error, the Secure Channel Session aborted and the Current Security Level reset to NO_SECURITY_LEVEL.
    - If a security error is found, the command shall be rejected with a security error, the Secure Channel Session aborted and the Current Security Level reset to NO_SECURITY_LEVEL.
    - In all other cases, the Secure Channel Session shall remain active and the Current Security Level unmodified. The Application is responsible for further processing the command.
- If a Secure Channel Session is aborted, it is still considered not terminated.
- The current Secure Channel Session shall be terminated (if aborted or still open) and the Current Security Level reset to NO_SECURITY_LEVEL on either:
    - Attempt to initiate a new Secure Channel Session (new INITIALIZE UPDATE command);
    - Termination of the Application Session (e.g. new Application selection);
    - Termination of the associated logical channel;
    - Termination of the Card Session (card reset or power off);
    - Explicit termination by the Application (e.g. invoking GlobalPlatform API).

6 Cryptographic Keys

6.1 AES Keys

Table 6-1: Security Domain Secure Channel Keys

Key	Usage	Length	Remark
Static Secure Channel Encryption Key (Key-ENC)	Generate session key for Decryption/Encryption (AES)	16, 24, 32 bytes	Mandatory
Static Secure Channel Message Authentication Code Key (Key-MAC)	Generate session key for Secure Channel authentication and Secure Channel MAC Verification/Generation (AES)	16, 24, 32 bytes	Mandatory
Data Encryption Key (Key-DEK)	Sensitive Data Decryption (AES)	16, 24, 32 bytes	Mandatory
Session Secure Channel Encryption Key (S-ENC)	Used for data confidentiality	Key-ENC length	Dynamically
Secure Channel Message Authentication Code Key for Command (S-MAC)	Used for data and protocol integrity	Key-MAC length	Dynamically
Secure Channel Message Authentication Code Key for Response (S-RMAC)	User for data and protocol integrity	Key-MAC length	Dynamically and Conditional

A Security Domain supporting Secure Channel Protocol '03', including the Issuer Security Domain, shall have at least one complete key set containing a Key-ENC, a Key-MAC, and a Key-DEK key having the same length. When loaded to the card, these keys should be encrypted with a key of same or higher strength. The card issuer may require that this recommendation be enforced depending on its security policy.

6.2 Cryptographic Usage

6.2.1 AES Session Keys

AES session keys shall be generated every time a Secure Channel is initiated and are used in the mutual authentication process. These same session keys may be used for subsequent commands if the Current Security Level indicates that secure messaging is required.

Session keys are generated to ensure that a different set of keys is used for each Secure Channel Session.

The session keys are derived from the static Secure Channel keys. The encryption key S-ENC is derived from Key-ENC. The Secure Channel MAC key S-MAC is derived from Key-MAC. Optionally (if the “i” parameter indicates R-MAC support), the Secure Channel R-MAC key S-RMAC is derived from Key-MAC. No AES session keys are generated for key and sensitive data encryption operations. That allows pre-processed data loading and simplifies the personalization process.

Key derivation shall use the data derivation scheme defined in section 4.1.5 with the following settings:

Table 6-2: AES Key Derivation Elements

Derived Session Key	Key K_I used in PRF	Derivation Constant (see Table 4-1)
S-ENC	Key-ENC	'04'
S-MAC	Key-MAC	'06'
S-RMAC	Key-MAC	'07'

The length of the session keys shall be reflected in the parameter “L” (i.e. '0080' for AES-128 keys, '00C0' for AES-192 keys and '0100' for AES-256 keys).

The “context” parameter shall be set to the concatenation of the host challenge (8 bytes) and the card challenge (8 bytes).

6.2.2 Challenges and Authentication Cryptograms

Both the card and the off-card entity (host) each generate a challenge and an authentication cryptogram. The off-card entity verifies the card cryptogram and the card verifies the host cryptogram. The cryptogram lengths shall be the same as the length of the challenges.

6.2.2.1 Card Challenge

As indicated in the “i” parameter (see Table 5-1), the card challenge shall either be random or pseudo-random.

If the SCP03 for a Security Domain is configured for pseudo-random challenge generation, the card challenge shall be calculated as follows:

- For each SCP03 key set, the Security Domain shall have one sequence counter of 3 bytes length. Whenever a key set is created or the whole key set is replaced by a single PUT KEY or STORE DATA command, the sequence counter shall be set to zero.
- Whenever a challenge generation is triggered by an INITIALIZE UPDATE command, the sequence counter shall be incremented and the new value shall be used in the calculation described below. When the maximum value is reached, the INITIALIZE UPDATE command shall be rejected with “conditions of use not satisfied”.
- The card challenge (8 bytes) is calculated using the data derivation scheme defined in section 4.1.5 with the static key Key-ENC and the derivation constant set to “card challenge generation” (i.e. '02'). The length of the challenge shall be reflected in the parameter “L” (i.e. '0040'). The “context” parameter shall be set to the concatenation of the sequence counter (3 bytes) and the AID of the application invoking the `SecureChannel` interface (5 to 16 bytes).

6.2.2.2 Card Authentication Cryptogram

The card cryptogram (8 bytes) is calculated using the data derivation scheme defined in section 4.1.5 with the session key S-MAC and the derivation constant set to “card authentication cryptogram generation”. The length of the cryptogram shall be reflected in the parameter “L” (i.e. '0040').

The “context” parameter shall be set to the concatenation of the host challenge (8 bytes) and the card challenge (8 bytes).

6.2.2.3 Host Authentication Cryptogram

The host cryptogram (8 bytes) is calculated using the data derivation scheme defined in section 4.1.5 with the session key S-MAC and the derivation constant set to “host authentication cryptogram generation”. The length of the cryptogram shall be reflected in the parameter “L” (i.e. '0040').

The “context” parameter shall be set to the concatenation of the host challenge (8 bytes) and the card challenge (8 bytes).

6.2.3 Message Integrity Using Explicit Secure Channel Initiation

SCP03 mandates the use of a MAC on the EXTERNAL AUTHENTICATE command.

For the EXTERNAL AUTHENTICATE command MAC verification, the “MAC chaining value” is set to 16 bytes '00'.

Once the cryptograms are successfully verified, the full 16-byte C-MAC of the previous command becomes the “MAC chaining value” for the subsequent C-MAC verification / R-MAC generation.

6.2.4 APDU Command C-MAC Generation and Verification

A C-MAC is generated by an off-card entity: it uses the S-MAC key and is applied across the MAC chaining value concatenated with the full APDU command being transmitted to the card including the header (5 bytes) and the data field in the command message. (It does not include Le.)

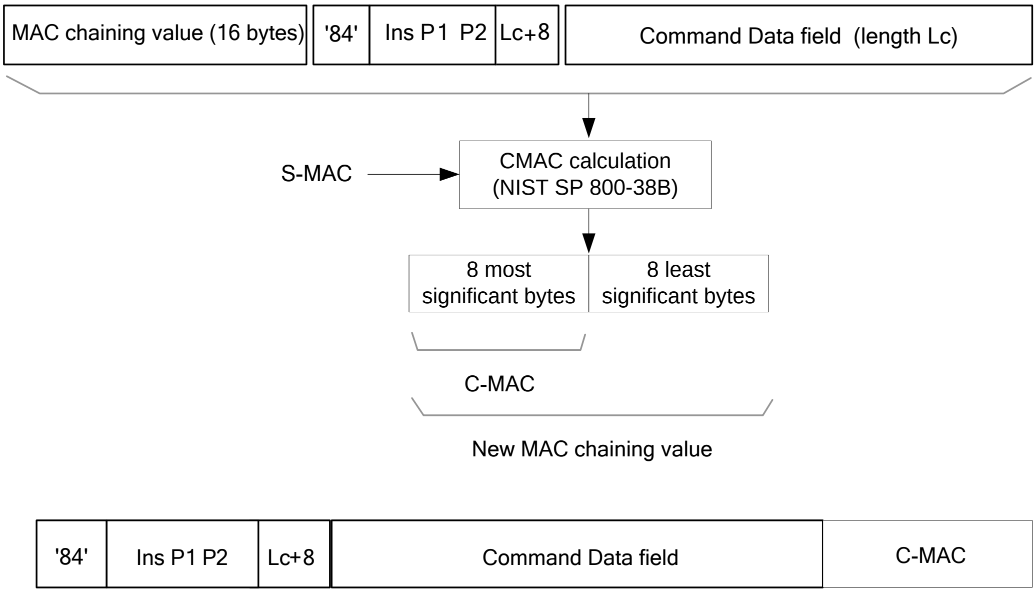
Modification of the APDU command header and padding is required prior to the MAC operation being performed.

The Secure channel shall support a MAC of 8 bytes length (even if the AES block length is 16 bytes). Hence the eight most significant bytes are considered.

The rules for APDU command header modification are as follows:

- The length of the command message (Lc) shall be incremented by 8 to indicate the inclusion of the C-MAC in the data field of the command message.
- The class byte shall be modified for the generation or verification of the C-MAC: The logical channel number shall be set to zero, bit 4 shall be set to 0 and bit 3 shall be set to 1 to indicate GlobalPlatform proprietary secure messaging. If the Secure Channel Session is occurring on a Supplementary Logical Channel, the class byte shall be modified after the C-MAC generation to indicate the logical channel number. If logical channel number 4 to 19 is used, the GlobalPlatform proprietary secure messaging is indicated by setting bit 6 to 1 – see [GPCS] Table 11-11 and Table 11-12. Conversely the logical channel number card is discarded and, if required, the secure messaging indication is adjusted for the verification. The logical channel number is not part of the integrity protection by the channel because is it established independently and outside of the scope of the Secure Channel establishment.

Figure 6-1: APDU C-MAC Generation


MAC chaining value (16 bytes)
'84'
Ins P1 P2
Lc+8
Command Data field (length Lc)
S-MAC
CMAC calculation (NIST SP 800-38B)
8 most significant bytes
8 least significant bytes
C-MAC
New MAC chaining value
'84'
Ins P1 P2
Lc+8
Command Data field
C-MAC

6.2.5 APDU Response R-MAC Generation and Verification

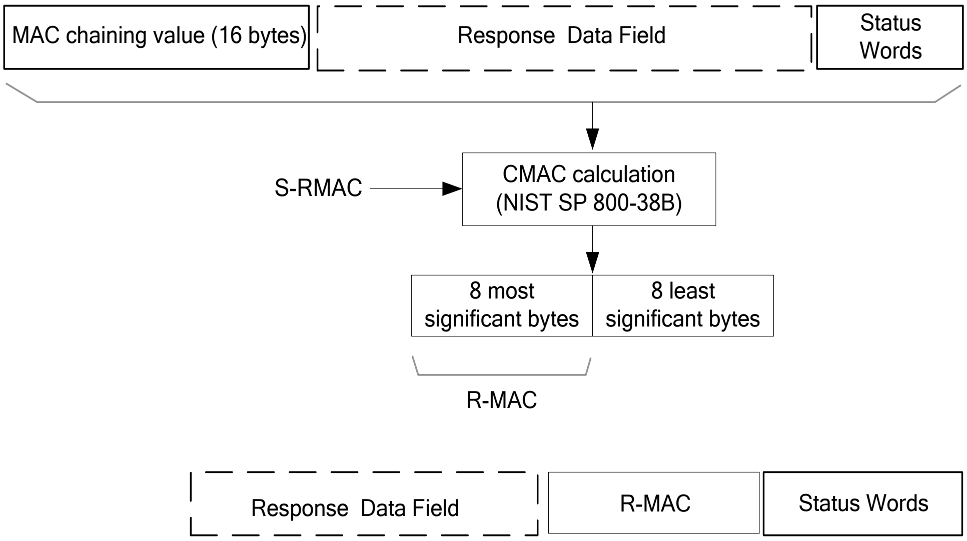
No R-MAC shall be generated and no protection shall be applied to a response that includes an error status word: in this case only the status word shall be returned in the response. All status words except '9000' and warning status words (i.e. '62xx' and '63xx') shall be interpreted as error status words.

The EXTERNAL AUTHENTICATE command/response doesn't return R-MAC.

The R-MAC is made of the first 8 bytes of the CMAC computed on the message made of the MAC chaining value, the response data field (if present) and the status bytes. The R-MAC calculation uses the S-RMAC key.

The R-MAC calculation is illustrated in Figure 6-2.

Figure 6-2: APDU R-MAC Generation

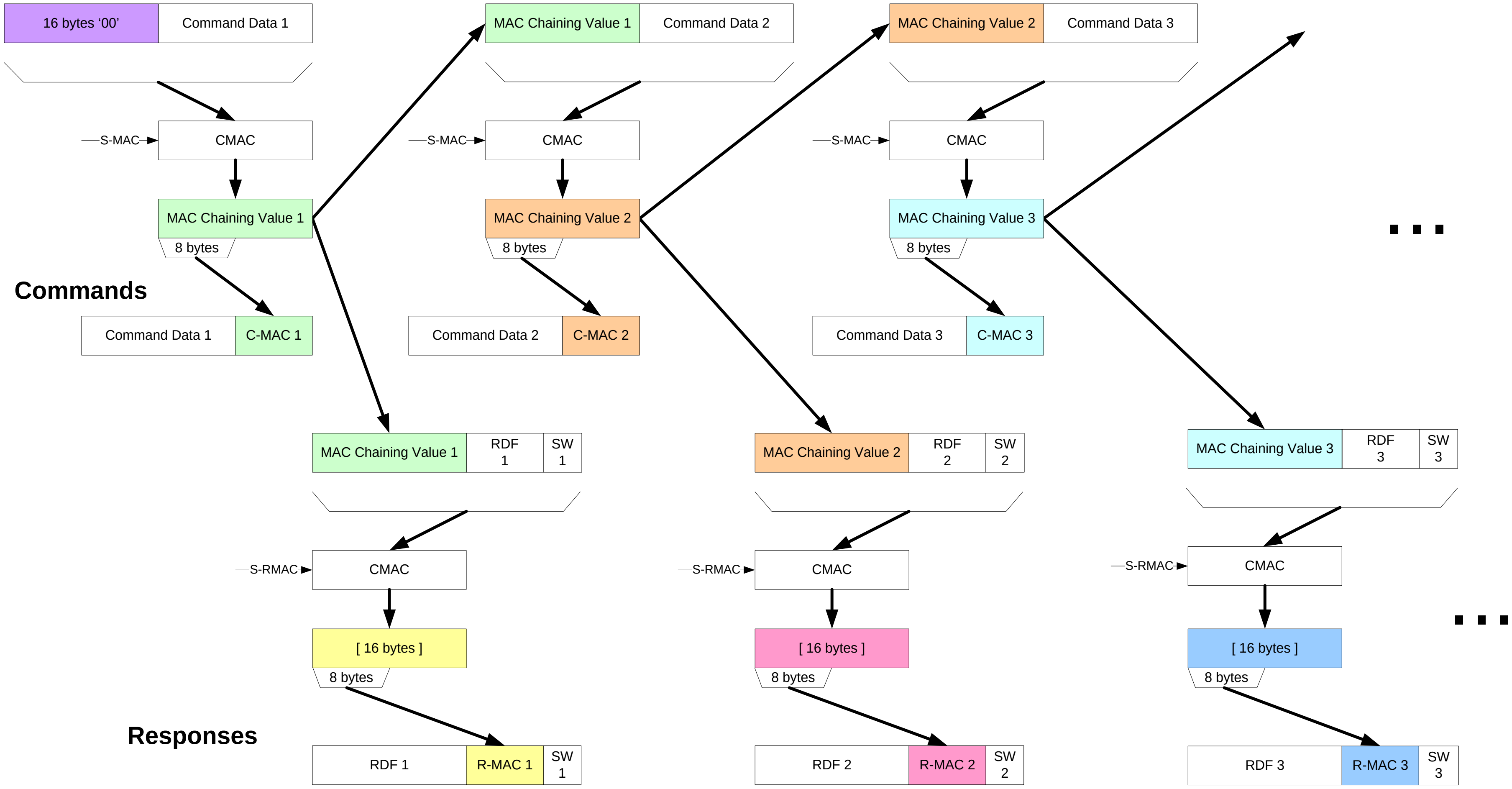


The off-card entity shall perform the same CMAC calculation on the response and use the same R-MAC session key employed by the card in order to verify the R-MAC.

The computed R-MAC becomes part of the response message.

Figure 6-3 illustrates the combined MAC chaining for command and responses.

Figure 6-3: MAC Chaining



Via the MAC changing value, consecutive commands are chained to each other, protecting their sequence.

Responses are linked to the respective command via the MAC changing value. As R-MAC uses a different session key than C-MAC, the same MAC chaining value can be used for the response and the next command. The scheme is adapted to the features of SCP03, where:

- R-MAC is optional, and
- R-MAC may be switched on and off during a secure channel session (see BEGIN/END R-MAC SESSION commands).

6.2.6 APDU Command C-MAC and C-DECRYPTION Generation and Verification

This section applies when both command confidentiality (C-DECRYPTION) and integrity (C-MAC) are required.

Depending on the security level defined in the initiation of the Secure Channel, all subsequent APDU commands within the Secure Channel may require secure messaging and such as use of a C-MAC (integrity) and encryption (confidentiality).

For each APDU command sent within the secure channel session, the off-card entity shall increment an encryption counter:

- The encryption counter's start value shall be set to 1 for the first command following a successful EXTERNAL AUTHENTICATE command.
- The encryption counter's binary value shall be left padded with zeroes to form a full block.
- This block shall be encrypted with S-ENC to produce the ICV for command encryption.

Note: This scheme fulfils the requirements described in [NIST 800-38A] for unpredictable ICVs when using CBC mode.

No encryption shall be applied to a command where there is no command data field. In this case, the encryption counter shall still be incremented as described above, and the message shall be protected as defined in section 6.2.4. Otherwise the off-card entity performs the process detailed hereafter.

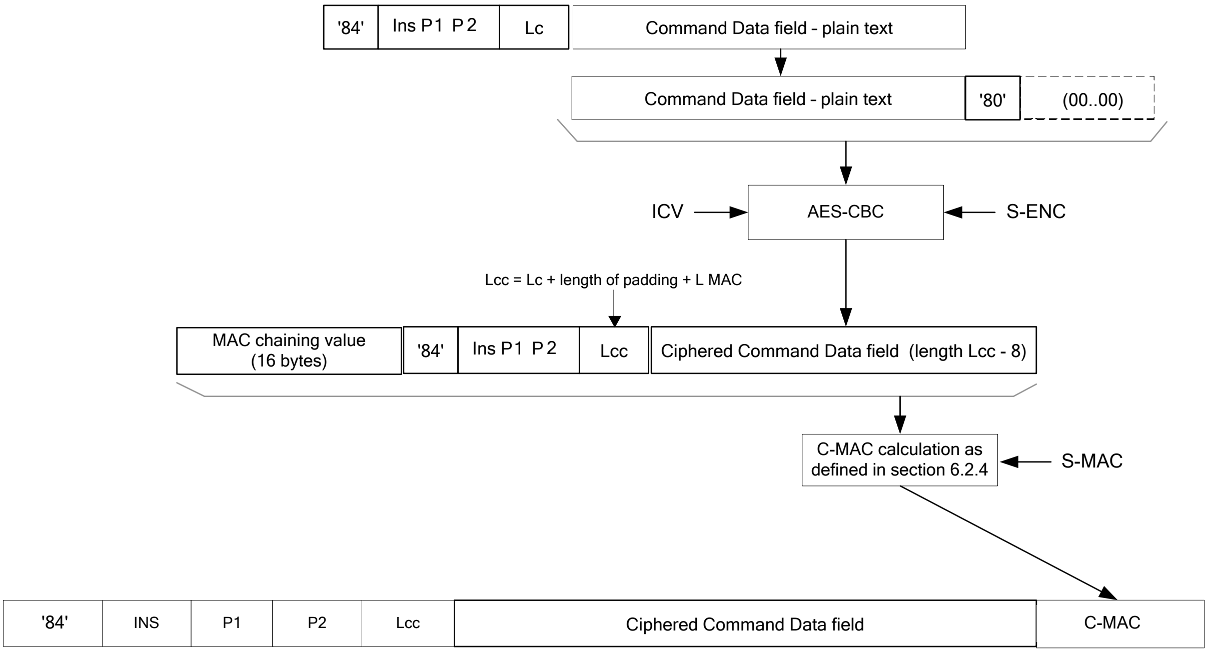
The off-card entity first encrypts the Command Data field and then computes the C-MAC on the command with the ciphered data field as described in section 6.2.4.

The command message encryption and decryption uses the Secure Channel encryption (S-ENC) session key and the AES encryption in CBC Mode. Prior to encrypting the data, the data shall be padded as defined in ~~section 4.1.4~~ [GPCS] section B.2.3. This padding becomes part of the data field.

The final Lc value (Lcc) is the sum of:

initial Lc + length of the padding + length of C-MAC

Figure 6-4: APDU Command Data Field Encryption


'84'
Ins P1 P2
Lc
Command Data field - plain text
Command Data field - plain text
'80'
(00..00)
ICV
AES-CBC
S-ENC
Lcc = Lc + length of padding + L MAC
MAC chaining value
(16 bytes)
'84'
Ins P1 P2
Lcc
Ciphered Command Data field (length Lcc - 8)
C-MAC calculation as
defined in section 6.2.4
S-MAC
'84'
INS
P1
P2
Lcc
Ciphered Command Data field
C-MAC

6.2.7 APDU Response R-MAC and R-ENCRYPTION Generation and Verification

This section applies when both response confidentiality (R-ENCRYPTION) and integrity (R-MAC) are required.

Depending on the security level defined in the initiation of the Secure Channel, all subsequent APDU responses within the Secure Channel may require secure messaging and such as use of an R-MAC (integrity) and encryption (confidentiality).

No encryption shall be applied to a response where there is no response data field: in this case the message shall be protected as defined in section 6.2.5. Otherwise the Card performs the process detailed hereafter.

The Card first encrypts the Response Data field and then computes the R-MAC on the response with the ciphered data field as described in section 6.2.5.

The response message encryption and decryption uses the Secure Channel encryption (S-ENC) session key and the AES encryption in CBC Mode. Prior to encrypting the data, the data shall be padded as defined in ~~section 4.1.4~~ [GPCS] section B.2.3. This padding becomes part of the data field.

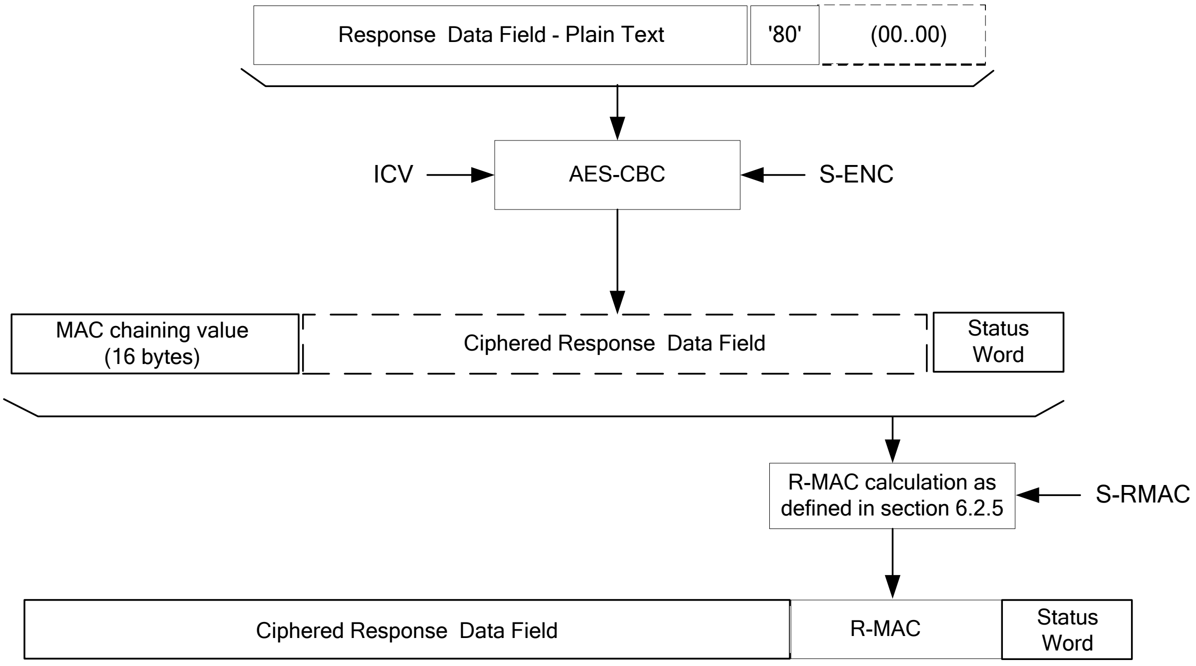
The ICV shall be calculated as follows:

- The padded counter block used for the generation of the ICV for command encryption shall also be used to generate the ICV for response encryption, however, with the following additional intermediate step: Before encryption, the most significant byte of this block shall be set to '80'.
- This block shall be encrypted with S-ENC to produce the ICV for response encryption. The modification in the most significant byte guarantees that the ICVs for R-ENCRYPTION are different from those used for C-DECRYPTION.

Note: This scheme fulfils the requirements described in [NIST 800-38A] for unpredictable ICVs when using CBC mode.

The final response APDU shall be the concatenation of the ciphered data, the R-MAC and the Status Word.

Figure 6-5: APDU Response Data Field Encryption


Response Data Field - Plain Text
'80'
(00..00)
ICV
AES-CBC
S-ENC
MAC chaining value (16 bytes)
Ciphered Response Data Field
Status Word
R-MAC calculation as defined in section 6.2.5
S-RMAC
Ciphered Response Data Field
R-MAC
Status Word

6.2.8 Key Sensitive Data Encryption and Decryption

Key data encryption is used when transmitting key sensitive data to the card and is over and beyond the security level required for the Secure Channel. For instance all AES keys transmitted to a card should be encrypted.

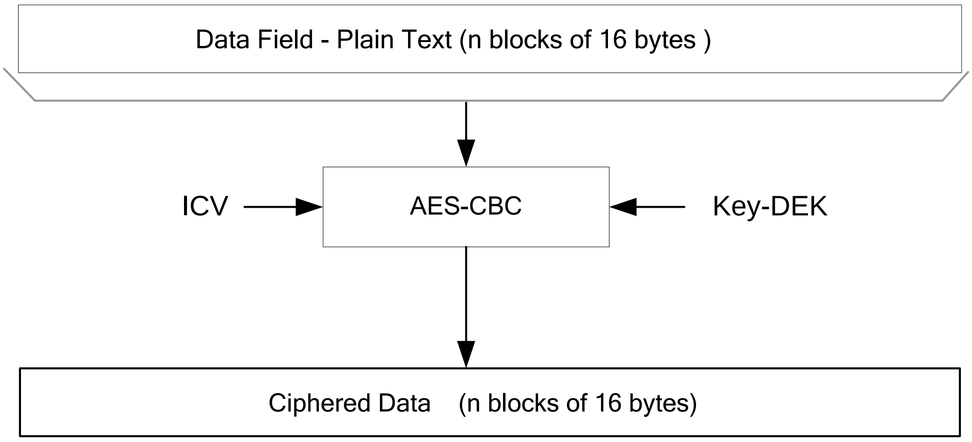
The Data encryption process uses the static data encryption key (Key-DEK) and the encryption method as described in ~~section 4.1.2~~ [GPCS] section B.2.1.

If the sensitive data to be encrypted are AES keys (16 or 32 bytes long), for instance for a PUT KEY command, then no padding is required for the data field prior to encryption as data block are multiple of 16 bytes long. For the 24-byte AES keys, padding of eight arbitrary bytes shall be appended prior to encryption. For the encryption of other keys see ~~section 7.2~~ [GPCS] section 11.8. For other sensitive data, padding is application specific and is out of scope of this document.

The AES CBC encryption with ICV set to zero is performed across the key sensitive data and the result of each encryption becomes part of the encrypted key data. This encrypted key data becomes part of the "clear text" data field in the command message.

The on-card decryption of key data is the exact opposite of the above operation.

Figure 6-6: Sensitive Data Encryption


Data Field - Plain Text (n blocks of 16 bytes)
ICV
AES-CBC
Key-DEK
Ciphered Data (n blocks of 16 bytes)

7 Commands

The following table presents the commands involved in Secure Channel Initiation and R-MAC Session Management.

Table 7-1: SCP03 Command Support

Command	Secure Channel Initiation
INITIALIZE UPDATE	✓
EXTERNAL AUTHENTICATE	✓
BEGIN R-MAC SESSION	
END R-MAC SESSION	

Ticks (✓) denote that support of the command is mandatory.

Blank cells denote that the support of the command is optional.

7.1 Secure Channel Commands

7.1.1 INITIALIZE UPDATE Command

Most of the following content was previously in [GPCS] Appendix D and referenced from this section. Exceptions are revision marked.

7.1.1.1 Definition and Scope

The INITIALIZE UPDATE command is used to transmit card and Secure Channel Session data between the card and the host. This command initiates the initiation of a Secure Channel Session.

At any time during a current Secure Channel, the INITIALIZE UPDATE command can be issued to the card in order to initiate a new Secure Channel Session.

7.1.1.2 Command Message

The INITIALIZE UPDATE command message is coded according to the following table:

Table 7-2: INITIALIZE UPDATE Command Message

Code	Value	Meaning
CLA	'80' - '83	See section 11.1.4
INS	'50'	INITIALIZE UPDATE
P1	'xx'	Key Version Number
P2	'xx'	Key Identifier: As per [GPCS] Table E-7 and section E.5.1.4, this parameter shall always have a value of '00'.
Lc	'08'	Length of host challenge
Data	'xx xx…'	Host challenge
Le	'00'	

7.1.1.3 Reference Control Parameter P1 – Key Version Number

The Key Version Number defines the Key Version Number within the Security Domain to be used to initiate the Secure Channel Session. If this value is zero, the first available key chosen by the Security Domain will be used.

If any of the mandatory keys listed in section 6.1 is missing in the targeted key set version, the INITIALIZE UPDATE command shall fail with error condition “Referenced data not found” as defined in [GPCS].

7.1.1.4 Reference Control Parameter P2 – Key Identifier

P2, Key Identifier, shall always be set to '00'. The Key Identifier together with the Key Version Number defined in reference control parameter P1 provide a unique reference to the set of keys to be used to initiate the Secure Channel Session.

7.1.1.5 Data Field Sent in the Command Message

The data field of the command message contains 8 bytes of host challenge. This challenge, chosen by the off-card entity, should be unique to this session.

7.1.1.6 Data Field Returned in the Response Message

The data field of the response message shall contain the concatenation without delimiters of the following data elements:

Table 7-3: INITIALIZE UPDATE Response Message

Name	Length	Presence
Key diversification data	10 bytes	Mandatory
Key information	3 bytes	Mandatory
Card challenge	8 bytes	Mandatory
Card cryptogram	8 bytes	Mandatory
Sequence Counter	3 bytes	Conditional

The key diversification data is data typically used by a backend system to derive the card static keys.

The key information includes the Key Version Number, the Secure Channel Protocol identifier, here '03', and the Secure Channel Protocol “i” parameter used in initiating the Secure Channel Session.

The card challenge is an internally generated random or pseudo random number.

The card cryptogram is an authentication cryptogram.

Sequence Counter is only present when SCP03 is configured for pseudo-random challenge generation.

The following content was previously in [GPCS] Appendix D and referenced from this section.

7.1.1.7 Processing State Returned in the Response Message

A successful execution of the command shall be indicated by status bytes '90' '00'.

This command may return either a general error condition as listed in section 11.1.3, General Error Conditions, or the following error condition:

Table 7-4: INITIALIZE UPDATE Error Condition

SW1	SW2	Meaning
'6A'	'88'	Referenced data not found

7.1.2 EXTERNAL AUTHENTICATE Command

The following content was previously in [GPCS] Appendix D and referenced from this section.

7.1.2.1 Definition and Scope

The EXTERNAL AUTHENTICATE command is used by the card to authenticate the host and to determine the level of security required for all subsequent commands.

A successful execution of the INITIALIZE UPDATE command shall precede this command.

7.1.2.2 Command Message

The EXTERNAL AUTHENTICATE command message is coded according to the following table:

Table 7-5: EXTERNAL AUTHENTICATE Command Message

Code	Value	Meaning
CLA	'84' - '87'	See [GPCS] section 11.1.4
INS	'82'	EXTERNAL AUTHENTICATE
P1	'xx'	Security level
P2	'00'	Reference control parameter P2
Lc	'10'	Length of host cryptogram and MAC
Data	'xx xx…'	Host cryptogram and MAC
Le		Not present

7.1.2.3 Reference Control Parameter P1 – Security Level

The reference control parameter P1 defines the level of security for all secure messaging commands following this EXTERNAL AUTHENTICATE command and within the Secure Channel Session.

Table 7-6: EXTERNAL AUTHENTICATE Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Description
0	0	1	1	0	0	1	1	C-DECRYPTION, R-ENCRYPTION, C-MAC, and R-MAC
0	0	0	1	0	0	1	1	C-DECRYPTION, C-MAC, and R-MAC
0	0	0	1	0	0	0	1	C-MAC and R-MAC
0	0	0	0	0	0	1	1	C-DECRYPTION and C-MAC
0	0	0	0	0	0	0	1	C-MAC
0	0	0	0	0	0	0	0	No secure messaging expected

The following content was previously in [GPCS] Appendix D and referenced from this section.

7.1.2.4 Reference Control Parameter P2

The reference control parameter P2 shall always be set to '00'.

7.1.2.5 Data Field Sent in the Command Message

The data field of the command message contains the host cryptogram and the APDU command MAC.

7.1.2.6 Data Field Returned in the Response Message

The data field of the response message is not present.

7.1.2.7 Processing State Returned in the Response Message

A successful execution of the command shall be indicated by status bytes '90' '00'.

This command may return either a general error condition as listed in [GPCS] section 11.1.3, General Error Conditions, or the following error condition:

Table 7-7: EXTERNAL AUTHENTICATE Error Condition

SW1	SW2	Meaning
'63'	'00'	Authentication of host cryptogram failed

7.1.3 BEGIN R-MAC SESSION Command

7.1.3.1 Definition and Scope

The BEGIN R-MAC SESSION command is used to initiate additional response security. The BEGIN R-MAC SESSION command may only be issued to the card within a secure channel. It may only be used to increase the security of the responses and only if command messages use at least the same security level.

The behavior of the implementation remains out of scope in the following case: This command is received and a BEGIN R-MAC SESSION command was previously received but no END ~~RMAC~~ R-MAC SESSION command was received in between.

7.1.3.2 Command Message

The BEGIN R-MAC SESSION command message is coded according to the following table:

Table 7-8: BEGIN R-MAC SESSION Command Message

Code	Value	Meaning
CLA	'80' - '87', 'C0' - 'CF', or 'E0' - 'EF'	Please refer to [GPCS] section 11.1.4
INS	'7A'	BEGIN R-MAC SESSION
P1	'xx'	Reference control parameter P1
P2	'01'	Reference control parameter P2
Lc	'XX'	Length of data field, if any
Data	'xx xx…'	BEGIN R-MAC SESSION data and C-MAC, if needed
Le		Not present

7.1.3.3 Reference Control Parameter P1

The reference control parameter P1 defines the level of security for all subsequent APDU response messages following this BEGIN R-MAC SESSION command (it does not apply to this command).

Table 7-9: BEGIN R-MAC SESSION Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Description
0	0	1	1	0	0	0	0	R-ENCRYPTION and R-MAC
0	0	0	1	0	0	0	0	R-MAC

When P1 is set to '10' each APDU response message during the R-MAC session includes an R-MAC. This setting may only be used if the secure channel session does not use R-MAC.

When P1 is set to '30' each APDU response message during the R-MAC session uses R-MAC and R-ENCRYPTION. This setting may only be used if the secure channel session does not use R-ENCRYPTION.

7.1.3.4 Reference Control Parameter P2

The reference control parameter P2 defines the beginning of the session for APDU response message integrity.

Table 7-10: BEGIN R-MAC SESSION Reference Control Parameter P2

b8	b7	b6	b5	b4	b3	b2	b1	Description
0	0	0	0	0	0	0	1	Begin R-MAC session

7.1.3.5 Data Field Sent in the Command Message

The data field of the BEGIN R-MAC SESSION contains an LV coded ‘data’ element and optionally a C-MAC. The card does not interpret the ‘data’. However since it is included in R-MAC calculation, this gives the off-card entity the possibility to include a challenge in the R-MAC.

7.1.3.6 Data Field Returned in the Response Message

If R-MAC was specified in the previous EXTERNAL AUTHENTICATE command, then the response to the BEGIN R-MAC Session shall contain an R-MAC. Otherwise, the data field of the response message is not present.

7.1.3.7 Processing State Returned in the Response Message

A successful execution of the command shall be indicated by status bytes '90' '00'.

The card will respond with a '6985' status word without aborting the secure channel in the following cases:

- If R-ENCRYPTION was specified in the previous EXTERNAL AUTHENTICATE command;
- If P1 is set to '10' and R-MAC was specified in the previous EXTERNAL AUTHENTICATE command;
- If C-MAC was not specified in the previous EXTERNAL AUTHENTICATE command;
- If P1 is set to '30' and C-DECRYPTION was not specified in the previous EXTERNAL AUTHENTICATE command.

This command may return a general error condition as listed in [GPCS] section 11.1.3, General Error Conditions.

7.1.4 END R-MAC SESSION Command

7.1.4.1 Definition and Scope

The END R-MAC SESSION command is used to terminate the additional response security that was initiated by the preceding BEGIN R-MAC SESSION command. The Secure Channel session returns to the Security Level established by the EXTERNAL AUTHENTICATE command that started the session. The END R-MAC SESSION command may be issued to the card at any time during an R-MAC session. If this command contains a wrong C-MAC, the entire Secure Channel session shall be aborted. In all other cases, if this command returns an error status word, the R-MAC session shall not be aborted. The R-MAC session shall be terminated if the secure channel is closed or the card is reset.

7.1.4.2 Command Message

The END R-MAC SESSION command message is coded according to the following table:

Table 7-11: END R-MAC SESSION Command Message

Code	Value	Meaning
CLA	'80' - '87', 'C0' - 'CF', or 'E0' - 'EF'	Please refer to [GPCS] section 11.1.4
INS	'78'	END R-MAC SESSION
P1	'00'	Reference control parameter P1
P2	'03'	Reference control parameter P2
Lc	'xx'	Length of data field, if any
Data	'xx xx…'	C-MAC, if needed
Le	'00'	

7.1.4.3 Reference Control Parameter P1

Reference control parameter P1 shall always be set to '00'.

7.1.4.4 Reference Control Parameter P2

The reference control parameter P2 is coded according to the following table:

Table 7-12: END R-MAC SESSION Reference Control Parameter P2

b8	b7	b6	b5	b4	b3	b2	b1	Description
0	0	0	0	0	0	1	1	End R-MAC session & return R-MAC

7.1.4.5 Data Field Sent in the Command Message

The data field of the command message may optionally contain a C-MAC.

7.1.4.6 Data Field Returned in the Response Message

The data field of the response message contains the R-MAC of the current R-MAC session.

7.1.4.7 Processing State Returned in the Response Message

A successful execution of the command shall be indicated by status bytes '90' '00'.

This command may return a general error condition as listed in [GPCS] section 11.1.3, General Error Conditions.

The content of the following sections has been moved to [GPCS]. The revision-marked deletions are omitted for readability.

7.2 PUT KEY Command (AES Key-DEK)

The content of this section has been moved to [GPCS] section 11.8.

7.3 STORE DATA (AES Key-DEK)

The content of this section has been moved to [GPCS] section 11.11.

8 AES for Card Content Management

The content of the following sections has been moved to [GPCS]. The revision-marked deletions are omitted for readability.

8.1 DAPs for AES

The content of this section has been moved to [GPCS] sections C.2 and C.3.

8.2 Tokens for AES

The content of this section has been moved to [GPCS] section C.4.

8.3 Receipts for AES

The content of this section has been moved to [GPCS] section C.5.