

Figure C-5: Make Selectable Token Calculation

Generating a Make Selectable Token ensures the following:

- Only the Application included in this signature and present in the Executable Load File may be made selectable on the card;
- That only the instance AID included in this signature may be used as the AID to select the instance;
- The Application may only be made selectable with the privileges included in this signature;
- Only the parameters included in this signature may be used to make the Application selectable;
- The issuer of the Make Selectable Token may only be the Security Domain Provider of the Security Domain with Token Verification Privilege.

The Make Selectable Token is an RSA signature of the following data that will be included in an INSTALL [for make selectable] command.

Name	Length
Reference control parameter P1	1
Reference control parameter P2	1
Length of the following data fields (including the length fields) - see note below	1
Executable Load File AID length (= '00')	1
Executable Module AID length (= '00')	1
Instance AID length	1
Instance AID	5-16
Privileges length	1
Privileges (byte 1 - byte 2 - byte 3)	1 or 3
Make Selectable parameters field length	1
Make Selectable parameters field	0-n

Table C-5: Data Elements Included in the Make Selectable Token

Note that 'Length of the following data fields' is the value of Lc of the INSTALL command, prior to a Token being added. It is coded on 1 byte with a value up to 255.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0201

C.4.4 Extradition Token

The Extradition Token allows for the verification of the extradition process. The token is a signature authorizing the extradition of an Application from one Security Domain to another. The OPEN shall request the Security Domain with Token Verification privilege to verify the Extradition Token.

The following figure shows Extradition Token generation.

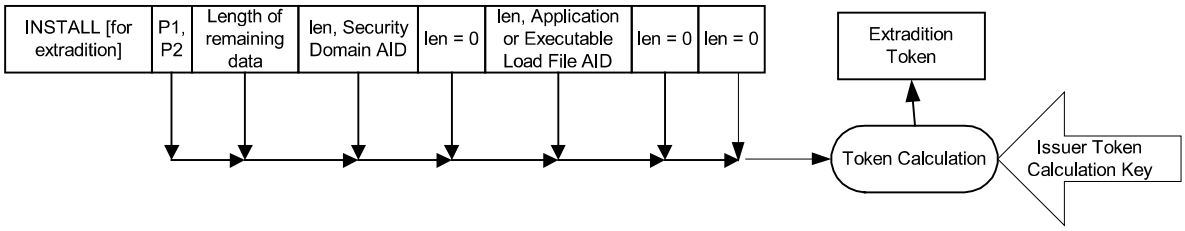


Figure C-6: Extradition Token Calculation

Generating an Extradition Token ensures the following:

- Only the Application or Executable Load File included in this signature may be extradited;
- The issuer of the Extradition Token may only be the Security Domain Provider of the Security Domain with Token Verification Privilege.

The Extradition Token is an RSA signature of the following data that will be included in an actual INSTALL [for extradition] command to be transmitted to the card.

Name	Length	Presence
Reference control parameter P1	1	Mandatory
Reference control parameter P2	1	Mandatory
Length of the following data fields (including the length fields) - see note below	1	Mandatory
Security Domain AID length	1	Mandatory
Security Domain AID	5-16	Conditional
Length = 0	1	Mandatory
Application or Executable Load File AID length	1	Mandatory
Application or Executable Load File AID	5-16	Mandatory
Length = 0	1	Mandatory
Length of Extradition Parameters field	1	Mandatory
Extradition Parameters field	0-n	Conditional

Table C-6: Data Elements Included in the Extradition Token

Note that 'Length of the following data fields' is the value of Lc of the INSTALL command, prior to a Token being added. It is coded on 1 byte with a value up to 255.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0202

C.4.5 Registry Update Token

The Registry Update Token allows for the verification of the registry update process. The token is a signature authorizing the updating of the GlobalPlatform Registry data for an Application, and optionally the extradition of an Application from one Security Domain to another. The OPEN shall request the Security Domain with Token Verification privilege to verify the Registry Update Token.

Figure C-7 shows the Registry Update Token generation.

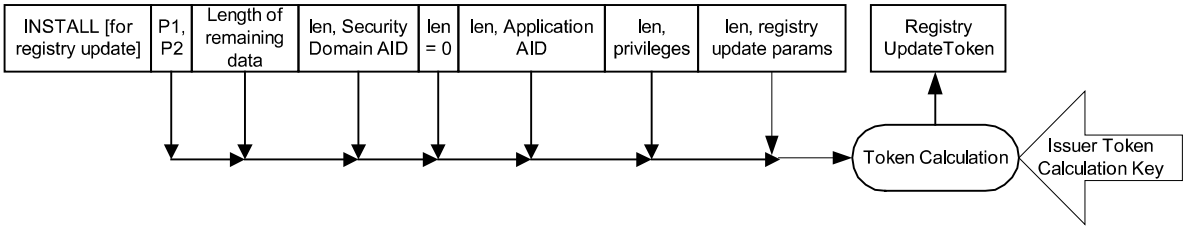


Figure C-7: Registry Update Token Calculation

Generating a Registry Update Token ensures the following:

- Only the Application included in this signature may have its GP Registry entry updated (and be extradited, in the case of extradition using registry update);
- The GP Registry entry of the Application may only be updated with the privileges and/or parameters included in this signature;
- The issuer of the Registry Update Token may only be the Security Domain Provider of the Security Domain with Token Verification Privilege

The Registry Update Token is an RSA signature of the following data that will be included in an actual INSTALL [for registry update] command to be transmitted to the card.

Name	Length	Presence
Reference control parameter P1	1	Mandatory
Reference control parameter P2	1	Mandatory
Length of the following data fields (including the length fields) - see note below	1	Mandatory
Security Domain AID length	1	Mandatory
Security Domain AID	0 or 5-16	Conditional
Length = 0	1	Mandatory
Application AID length	1	Mandatory
Application AID	5-16	Conditional
Length of Privileges	1	Mandatory
Privileges (byte 1 - byte 2 - byte 3)	0-3	Conditional
Length of Registry Update Parameters	1	Mandatory
Registry Update Parameters	0-n	Conditional

Table C-7: Data Elements Included in the Registry Update Token

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0203

Note that 'Length of the following data fields' is the value of Lc of the INSTALL command, prior to a Token being added. It is coded on 1 byte with a value up to 255.

C.4.6 Delete Token

The Delete Token is a signature authorizing the deletion of card content from the card. It allows for the verification of the card content removal process The Token is a signature of selected fields within the DELETE command APDU and is appended to the DELETE command.

Figure C-8 shows the Delete Token generation.

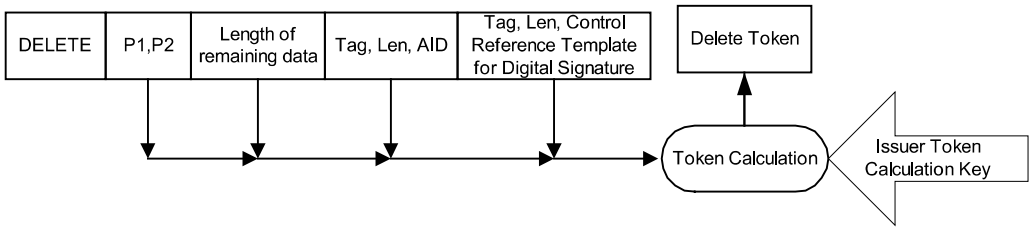


Figure C-8: Delete Token Calculation

Generating a Delete Token ensures the following:

- Only the Application or Executable Load File included in this signature may be deleted;
- The issuer of the Delete Token may only be the Security Domain Provider of the Security Domain with Token Verification Privilege.

The Delete Token is an RSA signature of the following data that will be included in the actual DELETE [card content] command to be transmitted to the card.

Name	Length
Reference control parameter P1	1
Reference control parameter P2	1
Length of the following data fields - see note below	1
Application or Executable Load File AID tag ('4F')	1
AID length	1
Application or Executable Load File AID	5-16
Control Reference Template for Digital Signature tag ('B6')	0 or 1
Control Reference Template for Digital Signature length	0-3
Control Reference Template for Digital Signature (Token) as defined in Table 11-22, comprising (in TLV format) Identification Number of the Security Domain with the Token Verification privilege (tag '42'), Image Number of the Security Domain with the Token Verification privilege (tag '45'), Application Provider id (tag '5F20') and Token identifier/number (tag '93')	0-n

Table C-8: Data Elements Included in the Delete Token

Note that 'Length of the following data fields' is the value of Lc of the DELETE command, prior to a Token being added. It is coded on 1 byte with a value up to 255.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0204

C.4.7 Load, Install and Make Selectable Token

The combined Load, Install and Make Selectable Token allows for the verification of the combined loading and installation process. The token is a signature authorizing the combined loading of an Executable Load File and installation to the card of an Application contained in a previously loaded file (Executable Load File). The OPEN shall request the Security Domain with Token Verification privilege to verify the combined Load, Install and Make Selectable Token.

Figure C-9 shows the combined Load, Install and Make Selectable Token generation.

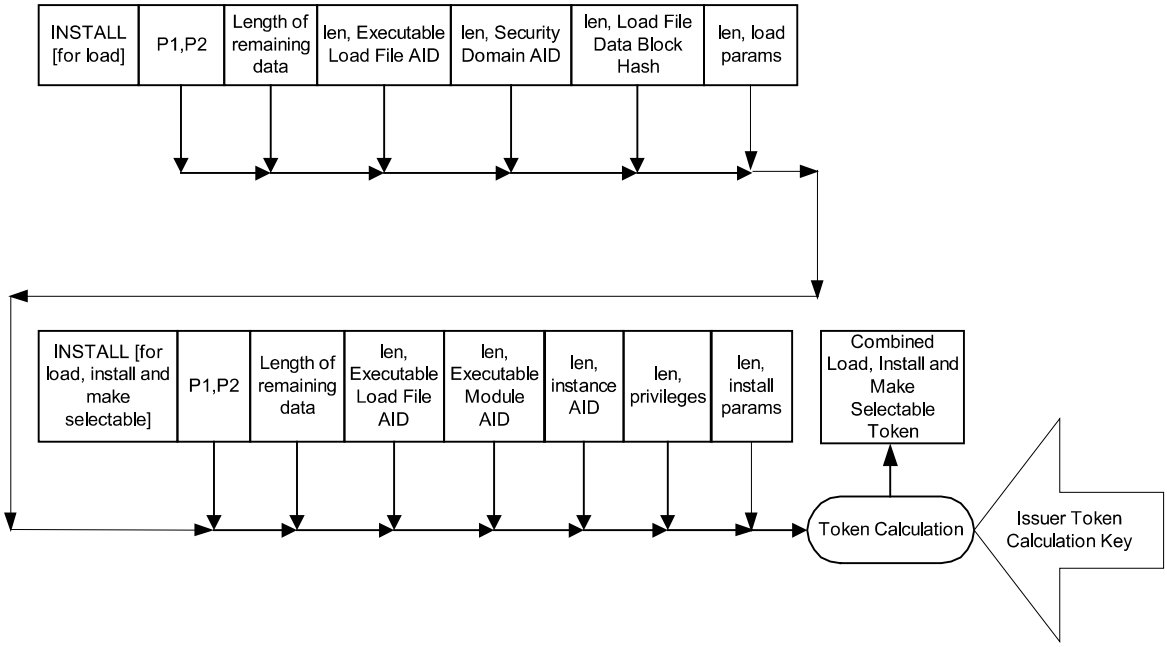


Figure C-9: Load, Install and Make Selectable Token Calculation

Generating a combined Load, Install and Make Selectable Token ensures the following:

- Only the application code included in this signature may be loaded to the card;
- The AID of the Load File Data Block may only be that included in the signature;
- The Executable Load File (derived from the Load File Data Block) and all Executable Modules within the Executable Load File may only be associated with the Security Domain included in the signature;
- Only the application (Executable Module) included in this signature and present in the Executable Load File may be installed on the card;
- That only the instance AID included in this signature may be used as the AID to select the instance;
- The Application may only be installed with the privileges included in this signature;
- Only the parameters included in this signature may be used to install the Application;

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0205

- The issuer of the combined Load, Install and Make Selectable Token may only be the Security Domain Provider of the Security Domain with Token Verification Privilege.

The combined Load, Install and Make Selectable Token is an RSA signature of the following data that will be included in an INSTALL [for load, install and make selectable] command.

Name	Length
Reference control parameter P1	1
Reference control parameter P2	1
Length of the following data fields (including the length fields)	1
Load File AID length	1
Load File AID	5-16
Security Domain AID length	1
Security Domain AID	0 or 5-16
Length of the Load File Data Block Hash	1
Load File Data Block Hash	0 or 20
Load parameters field length	1
Load parameters field	Var
Reference control parameter P1	1
Reference control parameter P2	1
Length of the following data fields (including the length fields)	1
Executable Load File AID length	1
Executable Load File AID	0 or 5-16
Executable Module AID length	1
AID within the Executable Load File	0 or 5-16
Instance AID length	1
Instance AID	5-16
Privileges length	1
Privileges (byte 1 - byte 2 - byte 3)	1 or 3
Install Parameters field length	1
Install Parameters (system and Application) field	Var

Table C-9: Data Elements Included in the Load, Install and Make Selectable Token

Note that 'Length of the following data fields' is the value of Lc of the INSTALL command, prior to a Token being added. It is coded on 1 byte with a value up to 255.

C.5 Receipts

Receipts are optional. They are generated by the Security Domain with Receipt Generation privilege, which requires knowledge of the appropriate receipt key. The Delete Receipt is comprised of data appropriate to the function that has been performed, together with Card Unique Data generated by the Security Domain with Receipt Generation privilege.

Each Receipt is generated either by:

- Symmetric cryptography using the receipt key, an ICV of binary zeroes and the signature method described in appendix B.1.2.2 - *Single DES Plus Final Triple DES MAC* across the receipt data.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0206

Prior to generating the signature, the data shall be padded according to appendix B.4 - *DES Padding*;

- asymmetric cryptography using the private key of the Security Domain with Receipt Generation privilege to decipher (sign) the SHA-1 digest of the receipt data using the signature method described in appendix B.2.1 - *Secure Hash Algorithm (SHA-1)*.

In addition to the appropriate cryptographic key, the Security Domain with Receipt Generation privilege also keeps track of a Confirmation Counter. The Confirmation Counter is a 16-bit value that is incremented by one (1) following each receipt generation. The Confirmation Counter is initialized to zero. When reaching its maximum value, the Confirmation Counter shall not be reset to zero. Cards are not required to support counter values beyond 32767.

C.5.1 Load Receipt

The Load Receipt provides confirmation that a successful load has occurred through the delegated loading process. A Load Receipt may be included in the response message for the last LOAD command of a sequence of LOAD commands.

Figure C-10 details the Load Receipt calculation.

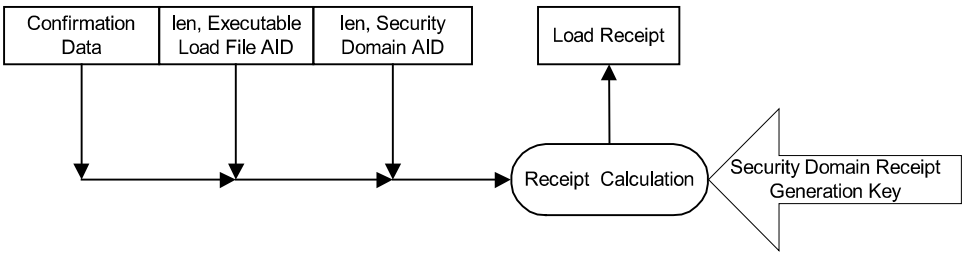


Figure C-10: Load Receipt Calculation

A Load Receipt is a signature of the following data:

Name	Length
Confirmation Data as defined in section 11.1.6	1-n
Executable Load File AID length	1
Executable Load File AID	5-16
Security Domain AID length	1
Security Domain AID	5-16

Table C-10: Data Elements Included in the Load Receipt

C.5.2 Install Receipt and Make Selectable Receipt

The Install/Make Selectable Receipt provides confirmation card that a successful installation/make selectable has occurred through the delegated installation/make selectable process. An Install/Make Selectable Receipt may be returned in the response message to the INSTALL [for install] command, the INSTALL [for make selectable] and the INSTALL [for install and make selectable] command. An Install Receipt is not returned in the response message to the INSTALL [for personalization] command.

Figure C-11 details the Install/Make Selectable Receipt calculation.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0207

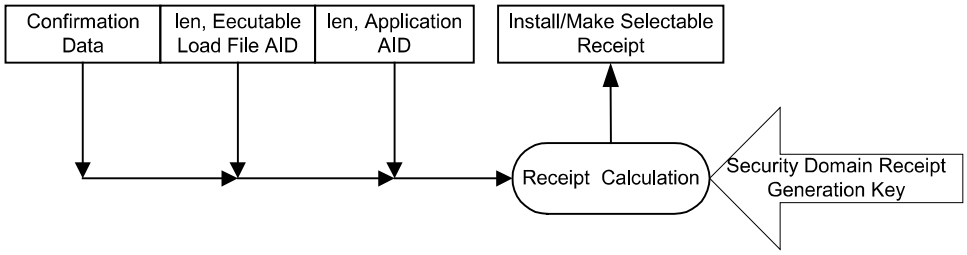


Figure C-11: Install/Make Selectable Receipt Calculation

An Install/Make Selectable Receipt is a signature of the following data:

Name	Length
Confirmation Data as defined in section 11.1.6	1-n
Executable Load File AID length	1
Executable Load File AID	5-16
Instance AID length indicator	1
Instance AID	5-16

Table C-11: Data Elements Included in the Install/Make Selectable Receipt

C.5.3 Extradition Receipt

The Extradition Receipt provides confirmation that the identified old Security Domain extradited the identified Application or Executable Load File to the identified new Security Domain, on a specific card, through the Delegated Management process. An Extradition Receipt may be returned in the response message to the INSTALL [for extradition] command.

Figure C-12 details the Extradition Receipt calculation performed by the Security Domain with Receipt Generation privilege.

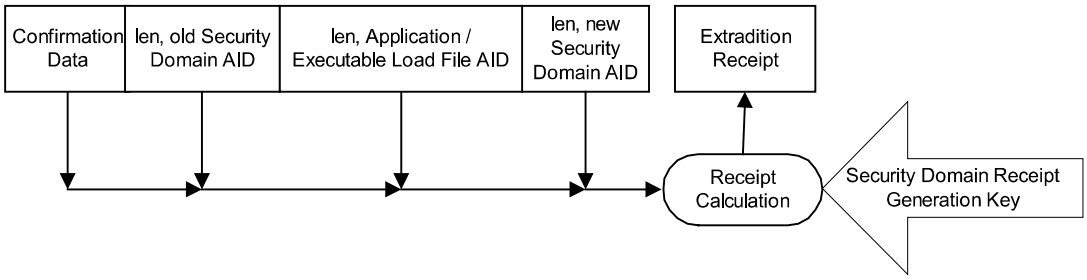


Figure C-12: Extradition Receipt Calculation

An Extradition Receipt is a signature of the following data:

Name	Length	Presence
Confirmation Data as defined in section 11.1.6	1-n	Mandatory
AID length indicator	1	Mandatory

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0208

Name	Length	Presence
Old Security Domain AID	5-16	Mandatory
AID length indicator	1	Mandatory
Application instance or Executable Load File AID	5-16	Mandatory
AID length indicator	1	Mandatory
New Security Domain AID	5-16	Mandatory

Table C-12: Data Elements Included in the Extradition Receipt

C.5.4 Registry Update Receipt

The Registry Update Receipt provides confirmation that GlobalPlatform Registry data (Privileges and/or Registry Update Parameters) for the identified Application was modified to the specified values on a specific card.

If registry update is used for extradition, the Registry Update Receipt also provides confirmation that the identified old Security Domain extradited the identified Application to the identified new Security Domain, on a specific card, through the Delegated Management process.

The following figure details the Registry Update Receipt calculation performed by the Security Domain with Receipt Generation privilege.

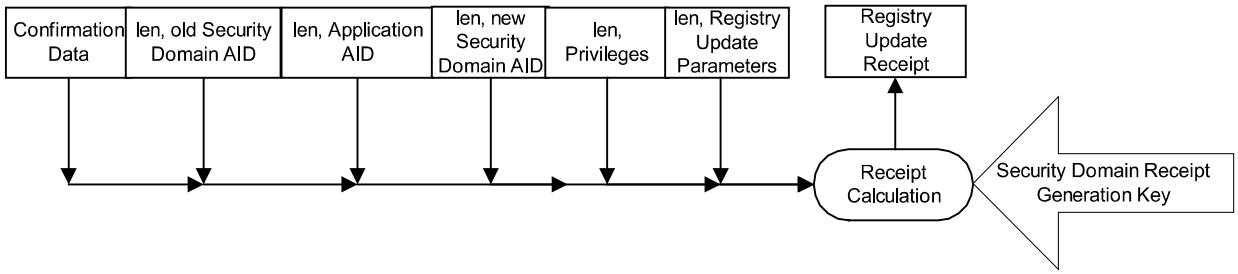


Figure C-13: Registry Update Receipt Calculation

A Registry Update Receipt is a signature of the following data:

Name	Length	Presence
Confirmation Data as defined in section 11.1.6	1-n	Mandatory
AID length indicator	1	Mandatory
Old Security Domain AID	0 or 5-16	Conditional
AID length indicator	1	Mandatory
Application AID	0 or 5-16	Conditional
AID length indicator	1	Mandatory
New Security Domain AID	0 or 5-16	Conditional
Length of Privileges	1	Mandatory
Privileges (byte 1 - byte 2 – byte 3)	0-3	Conditional
Length of Registry Update Parameters	1	Mandatory

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0209

Name	Length	Presence
Registry Update Parameters	0-n	Conditional

Table C-13: Data Elements Included in the Registry Update Receipt

The presence of conditional data depends on what Registry data (Privileges and/or Registry Update Parameters) is being updated and whether registry update is being used for extradition. See sections 11.5.2.3.5 and 11.5.2.3.7 for further details and the format of data. Security Domain AIDs are present for extradition.

C.5.5 Delete Receipt

The Delete Receipt provides proof that the identified Security Domain deleted an Application, an Executable Load File, or an Executable Load File and all its Application instances from a specific card through the delegated deletion process. The Delete Receipt may be returned in the response message to the DELETE [card content] command.

Figure C-14 details the Delete Receipt calculation performed by the Security Domain with Receipt Generation privilege.

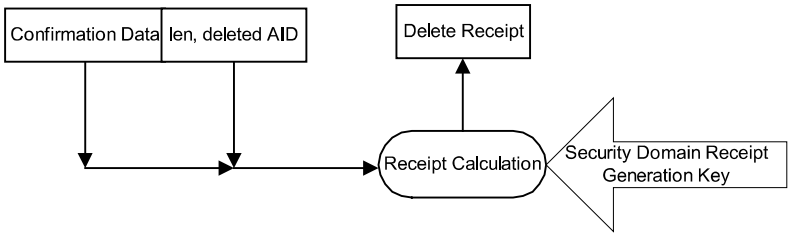


Figure C-14: Delete Receipt Calculation

A Delete Receipt is a signature of the following data:

Name	Length
Confirmation Data as defined in section 11.1.6	1-n
AID length indicator	1
Application instance or Executable Load File AID	5-16

Table C-14: Data Elements Included in the Delete Receipt

If the function was to delete an Executable Load File and all its Application instances, the receipt returns only the Executable Load File AID.

C.5.6 Combined Load, Install and Make Selectable Receipt

The combined Load, Install and Make Selectable Receipt provides confirmation card that a successful installation has occurred through the delegated installation process. A combined Load, Install and Make Selectable Receipt may be returned in the response message to the INSTALL [for load, install and make selectable] command.

Figure C-15 details the combined Load, Install and Make Selectable Receipt calculation.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0210

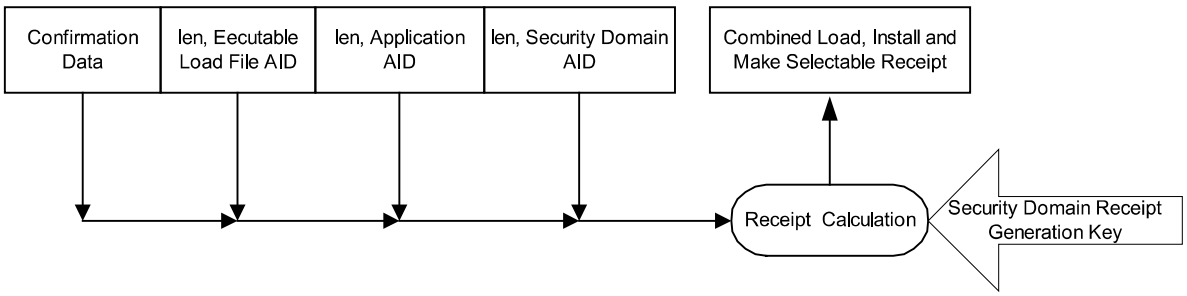


Figure C-15: Load, Install and Make Selectable Receipt Calculation

A combined Load, Install and Make Selectable Receipt is a signature of the following data:

Name	Length
Confirmation Data as defined in section 11.1.6	1-n
Executable Load File AID length	1
Executable Load File AID	5-16
Instance AID length indicator	1
Instance AID	5-16
Security Domain AID length	1
Security Domain AID	5-16

Table C-15: Data Elements Included in the Load, Install and Make Selectable Receipt

C.6 DAP Verification

Load File Data Block Signature verification is used to ensure that the Application has been validated.

In order to support DAP Verification, a Security Domain with DAP Verification privileges shall be present on the card. The Security Domain shall have knowledge of the appropriate key.

C.6.1 PKC Scheme

An entity generates a signature of the Load File Data Block Hash as defined in appendix B.2.1 - *Secure Hash Algorithm (SHA-1)*, i.e. computing a SHA-1 message digest of the Load File Data Block Hash and applying the DAP Verification private key to obtain the signature. This RSA signature is the decryption with the DAP Verification private key of the SHA-1 message digest of the Load File Data Block Hash (i.e. hash of a hash value). The signature scheme for DAP verification is as defined in appendix B.3. The signature is provided to any entity responsible for loading the signed application code to a GlobalPlatform card. The AID of the Security Domain verifying the DAP along with this signature are placed in the Load File as a DAP Block to be transmitted to the card.

C.6.2 DES Scheme

A Security Domain's provider generates a signature of the Load File Data Block Hash. This signature is a MAC of the Load File Data Block Hash according to appendix B.1.2.2 - *Single DES Plus Final Triple DES*. Padding of the data is as defined in appendix B.4 - *DES Padding*. The signature is provided to any entity responsible for loading the signed application code to a GlobalPlatform card. The AID of the Security

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0211

Domain verifying the DAP along with this signature are placed in the Load File as a DAP Block to be transmitted to the card.

C.7 GlobalPlatform on MULTOS

C.7.1 Keys

In order to perform Card Content management operations a GlobalPlatform on MULTOS smart card requires the following key:

- Issuer's key.

The following keys may optionally be supplied to the card during a card content loading operation:

- Application Provider's key;
- Application Provider's transport keys.

These keys are described in MAO-DOC-REF-009.

C.7.2 Cryptographic Structures

A GlobalPlatform on MULTOS smart card utilizes the following cryptographic structures:

- Public Key Certificate;
- Card Content Management Certificates:
  - Application Load Certificate;
  - Application Delete Certificate.
- Application Provider's Signature;
- Key Transformation Unit.

These structures are described in MAO-DOC-REF-009.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0212

D Secure Channel Protocol '01' (Deprecated)

This Secure Channel Protocol is deprecated, and will be removed from a future version of the Specification. Use of another Secure Channel Protocol, such as Secure Channel Protocol '02' (SCP02), is recommended.

D.1 Secure Communication

D.1.1 SCP01 Secure Channel

The 3 levels of security provided by SCP01 are:

Mutual authentication - in which the card and the off-card entity each prove that they have knowledge of the same secrets;

Integrity and data origin authentication - in which 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 - in which data being transmitted from the off-card entity to the card, is not viewable by an unauthorized entity.

A further level of security applies to sensitive data (e.g. secret keys) that shall always be transmitted as confidential data.

A maximum of 3 Supplementary Logical Channels is supported in SCP01.

Only explicit Secure Channel initiation is supported in SCP01. Both implicit and explicit Secure Channel termination are supported in SCP01.

In SCP01 the card shall support the following implementation options defined by "i" (see appendix H - *GlobalPlatform Data Values and Card Recognition Data*):

- "i" = '05': Initiation mode explicit, C-MAC on modified APDU, ICV set to zero, no ICV encryption, 3 Secure Channel Keys;
- "i" = '15': Initiation mode explicit, C-MAC on modified APDU, ICV set to zero, ICV encryption, 3 Secure Channel Keys.

Implementation option '15' is an enhancement over implementation option '05' and is therefore recommended.

D.1.2 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 generated.

The Secure Channel is explicitly initiated by the off-card entity using the INITIALIZE UPDATE and EXTERNAL AUTHENTICATE commands defined in this section. 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 appendix D.4.2 - *EXTERNAL AUTHENTICATE Command*) as to what level of security is required for the current Secure

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0213

Channel (integrity and/or confidentiality) and apply this level of security to all the 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 the 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 appendix D.4.1 - *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 appendix D.3.1 - *DES Session Keys*).

This card cryptogram along with 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 appendix D.4.2 - *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.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0214

SCP01 Mutual Authentication Flow

The following flow is an example of mutual authentication between a card and an off-card entity. This flow shows mutual authentication occurring between a Security Domain and an off-card entity.

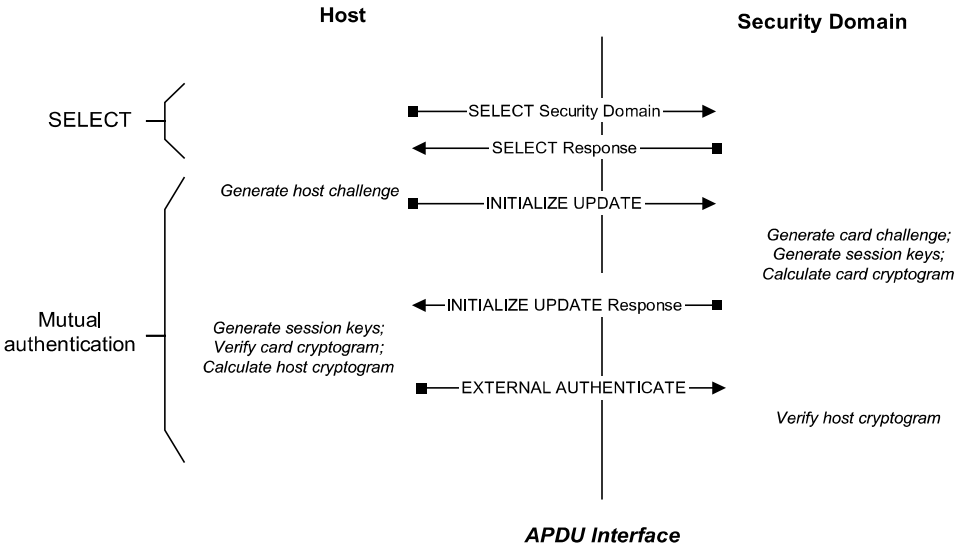


Figure D-1: Mutual Authentication Flow (Security Domain)

Expanding the authentication process shown in the flow described in section 7.3.4 - *Runtime Messaging Support*, it can be seen how an Application would use the services of an associated Security Domain to achieve mutual authentication.

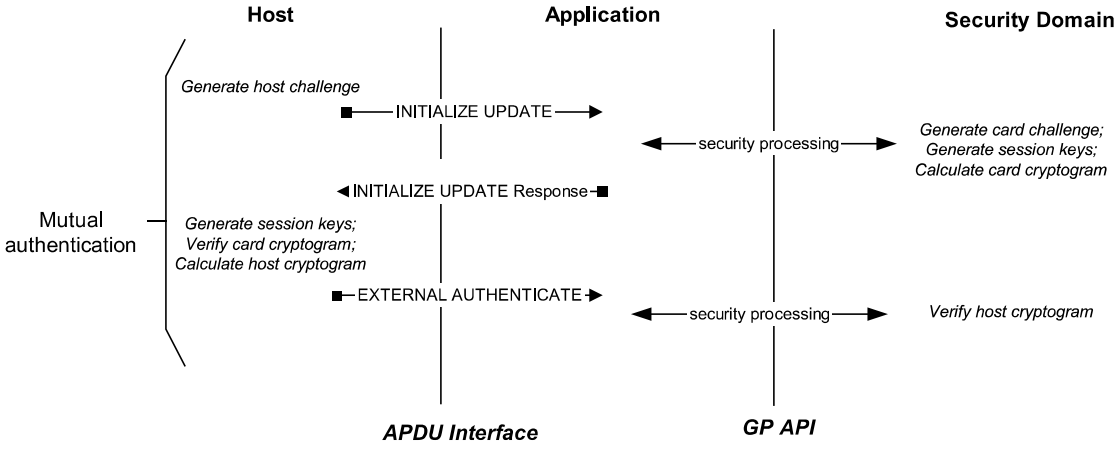


Figure D-2: Mutual Authentication Flow (using services of Security Domain)

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0215

D.1.3 Message Integrity

The C-MAC is generated by applying multiple chained DES operations (using a Secure Channel 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 before encryption.)

The integrity of the sequence of commands being transmitted to the card is achieved by using the C-MAC from the current command as the (possibly encrypted) Initial Chaining Vector (ICV) for the subsequent command. This ensures the card that all commands in a sequence have been received.

D.1.4 Message Data Confidentiality

The message data field is encrypted by applying multiple chained DES operations (using a Secure Channel session key generated during the mutual authentication process) across the entire data field of the command message to be transmitted to the card, regardless of its contents (clear text data and/or already protected sensitive data).

D.1.5 ICV Encryption

As an enhancement to the C-MAC mechanism, the ICV is encrypted before being applied to the calculation of the next C-MAC. The encryption mechanism used is triple DES with the C-MAC session key. The first ICV of a session, used to generate the C-MAC on the EXTERNAL AUTHENTICATE command, is not encrypted.

D.1.6 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 '01', the Current Security Level established in a Secure Channel Session is a bitmap combination of the following values: AUTHENTICATED, C_MAC, and C_DECRYPTION. 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 a successful processing of an EXTERNAL AUTHENTICATE command: AUTHENTICATED shall be cleared once the Secure Channel Session is terminated;
- AUTHENTICATED and C_MAC after a successful processing of an EXTERNAL AUTHENTICATE command with P1 indicating C-MAC (P1='01'): AUTHENTICATED and C-MAC shall be cleared once the Secure Channel Session is terminated;
- AUTHENTICATED, C_MAC, and C_DECRYPTION after a successful processing of an EXTERNAL AUTHENTICATE command with P1 indicating C-MAC and command encryption (P1= '03'): AUTHENTICATED, C_MAC, and C_DECRYPTION shall be cleared once the Secure Channel Session is terminated.

D.1.7 Protocol Rules

In accordance with the general rules described in chapter 10 - *Secure Communication*, the following protocol rules apply to Secure Channel Protocol ‘01’:

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0216

- 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 equal 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 the Security Domain supports application data encryption and/or decryption, it shall decrypt or encrypt a block of secret data upon request. If the service is not supported or if (one of) the appropriate cryptographic key(s) is not available, the request shall be rejected but the Current Security Level, Session Security Level and Secure Channel Session in operation (if any) shall not be impacted;
- 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).

D.2 Cryptographic Keys

Key	Usage	Length	Remark
Secure Channel Encryption Key (S-ENC)	Secure Channel Authentication & Encryption (DES)	16 bytes	Mandatory
Secure Channel Message Authentication Code Key (S-MAC)	Secure Channel MAC Verification (DES)	16 bytes	Mandatory

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0217

Key	Usage	Length	Remark
Data Encryption Key (DEK)	Sensitive Data Decryption (DES)	16 bytes	Mandatory

Table D-1: Security Domain Secure Channel Keys

A Security Domain, including the Issuer Security Domain shall have at least one key set containing 3 keys to be used in the initiation and use of a Secure Channel. These keys are all double length DES keys and are the following:

- The Secure Channel encryption key (S-ENC) and the Secure Channel MAC key (S-MAC). These keys are only used to generate Secure Channel session keys during the initiation of a Secure Channel;
- The data encryption key (DEK) for decrypting sensitive data, e.g. secret or private keys. This key is a double length DES key and is used as a static key.

D.3 Cryptographic Usage

D.3.1 DES Session Keys

DES 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. While this is not required for key encryption operations, it is important for authentication operations, MAC generation and verification and command message encryption and decryption. It is therefore only necessary to create session keys from the static Secure Channel encryption key (S-ENC) and the Secure Channel MAC key (S-MAC).

DES session keys are created using the static Secure Channel encryption key (S-ENC) and the Secure Channel MAC key (S-MAC) and the random host and card challenges. Creating session keys involves 3 steps.

- Generating the session key derivation data. (The same derivation data is used to create both the Secure Channel encryption and Secure Channel MAC session keys.);
- Creating the Secure Channel encryption session key;
- Creating the Secure Channel MAC session key.

The DES operation used to generate these keys is always triple DES in ECB mode.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0218

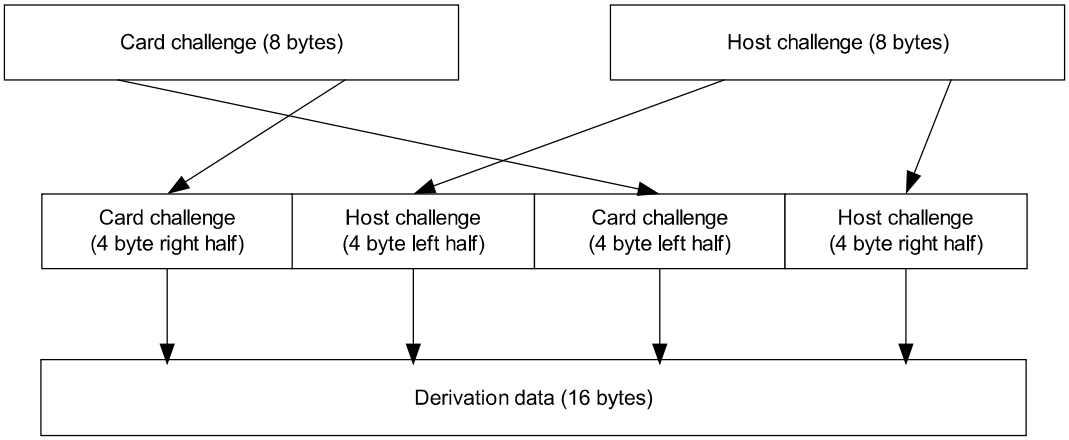


Figure D-3: Session Key - Step 1 - Generate Derivation data

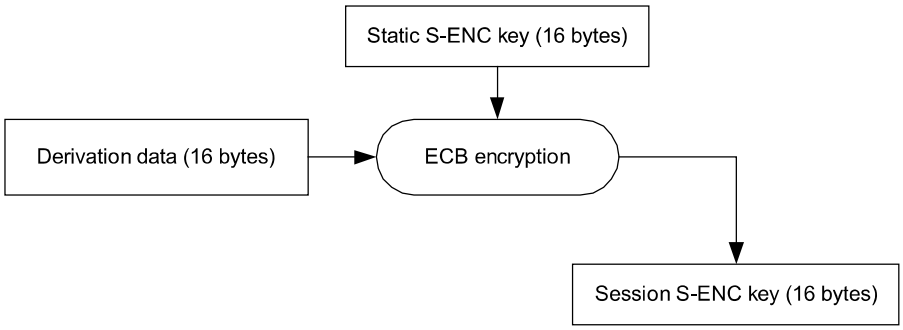


Figure D-4: Session Key - Step 2 - Create S-ENC Session Key

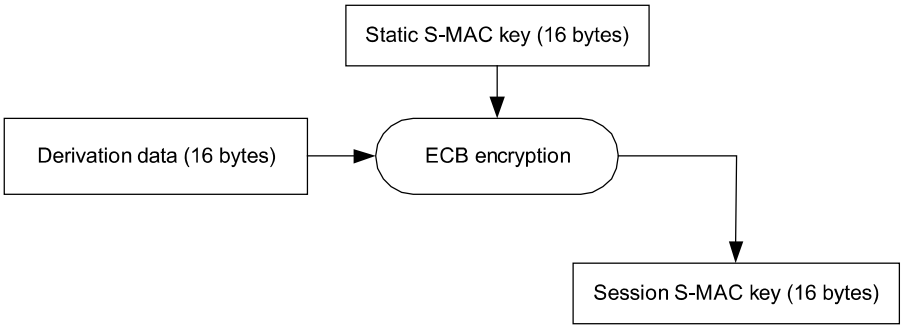


Figure D-5: Session Key – Step 3 - Create S-MAC Session Key

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0219

D.3.2 Authentication Cryptograms

Both the card and the off-card entity (Host) generate an authentication cryptogram. The off-card entity verifies the card cryptogram and the card verifies the host cryptogram. Generating or verifying an authentication cryptogram uses the S-ENC session key and the signing method described in appendix B.1.2.1 - *Full Triple DES*.

D.3.2.1 Card Authentication Cryptogram

The generation and verification of the card cryptogram is performed by concatenating the 8-byte host challenge and 8-byte card challenge resulting in a 16-byte block.

Applying the same padding rules defined in appendix B.4 - *DES Padding*, the data shall be padded with a further 8-byte block ('80 00 00 00 00 00 00 00').

The signature method, using the S-ENC session key and an ICV of binary zeroes, is applied across this 24-byte block and the resulting 8-byte signature is the card cryptogram.

D.3.2.2 Host Authentication Cryptogram

The generation and verification of the host cryptogram is performed by concatenating the 8-byte card challenge and 8-byte host challenge resulting in a 16-byte block.

Applying the same padding rules defined in appendix B.4 - *DES Padding*, the data shall be padded with a further 8-byte block ('80 00 00 00 00 00 00 00').

The signature method, using the S-ENC session key and an ICV of binary zeroes, is applied across this 24-byte block and the resulting 8-byte signature is the host cryptogram.

D.3.3 APDU Command MAC Generation and Verification

The Secure Channel mandates the use of a MAC on the EXTERNAL AUTHENTICATE command. Depending on the Session Security Level defined in the initiation of the Secure Channel, all other commands within the Secure Channel may require secure messaging and as such the use of a C-MAC.

A C-MAC is generated by an off-card entity and applied across 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 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 to indicate that this APDU command includes secure messaging. This is achieved by setting bit b3 of the class byte. For all the commands defined in this specification, the class byte of commands that contain secure messaging shall be '84'. The C-MAC generation and verification does not reflect the logical channel information included in the class byte of the command: the logical channel number is always assumed to be zero in the generation or verification of the C-MAC;
- The rules for MAC padding are as defined in appendix B.4 - *DES Padding*.

As the ICV is used to chain the commands for command sequence integrity, the value of the ICV depends on which APDU command within the sequence the MAC is being generated for:

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0220

- For the EXTERNAL AUTHENTICATE command, the ICV is set to binary zeroes;
- For any command following the EXTERNAL AUTHENTICATE command, the ICV is the C-MAC value successfully verified for the previous command received by the card. (Prior to using the ICV, the ICV can be encrypted as described in appendix D.1.5 - ICV Encryption.)

The signature method defined in appendix B.1.2.1 - *Full Triple DES*, using the MAC session key and the ICV, is applied across the padded data block and the resulting 8-byte signature is the C-MAC.

The C-MAC is appended at the end of the APDU command message excluding any padding but including the modifications made to the command header (class and Lc).

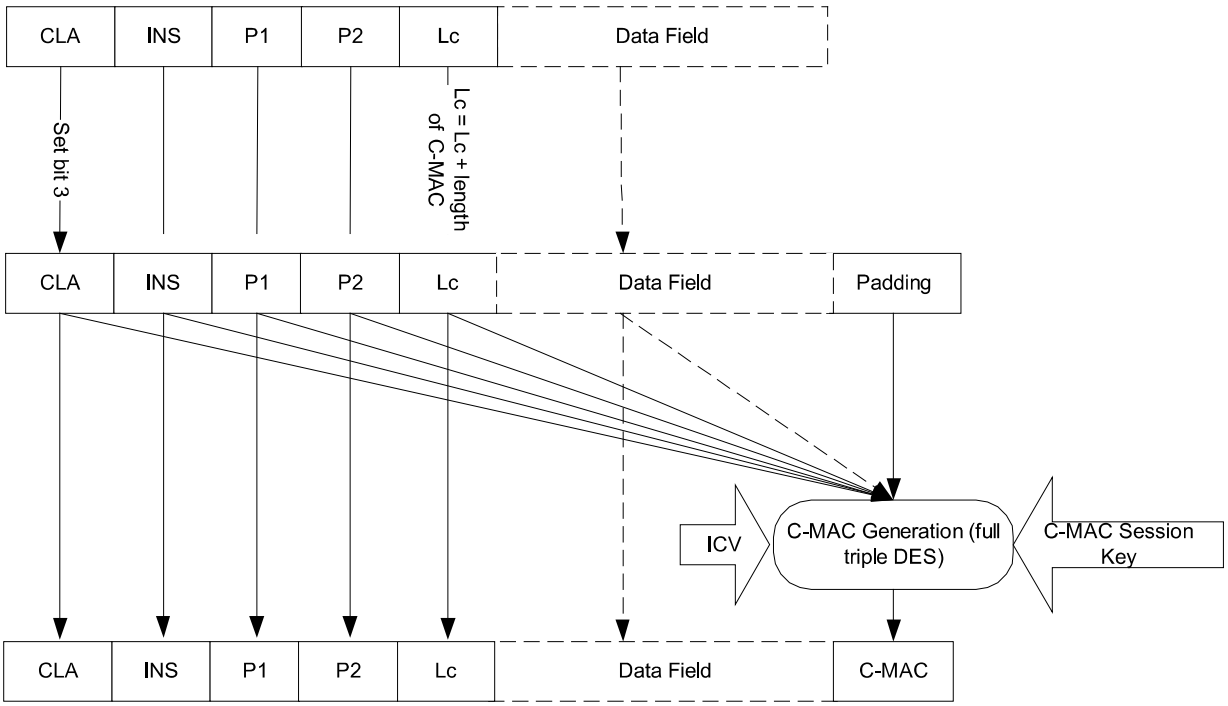


Figure D-6: APDU Command MAC Generation and Verification

If no other secure messaging is required, the message is now prepared for transmission to the card. The C-MAC shall be retained and used as the ICV for the subsequent C-MAC verification (if any).

The card, in order to verify the C-MAC, shall perform the same padding mechanism to the data and use the same ICV and MAC session key employed by the off-card entity in order to verify the C-MAC. As with the off-card entity, the C-MAC shall be retained and used as the ICV for the subsequent C-MAC verification (if any).

If the Secure Channel Session is occurring on a Supplementary Logical Channel, the logical channel number is added to the class byte of the message after the C-MAC has been generated and prior to the message being transmitted to the card. The card shall ignore any logical channel information in the class byte (i.e. assume the logical channel number is zero) prior to verifying the C-MAC.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0221

D.3.4 APDU Data Field Encryption and Decryption

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

If confidentiality is required, the off-card entity encrypts the “clear text” data field of the command message being transmitted to the card. (If the APDU command does not originally contain command data, encryption is not performed.) This excludes the header and the C-MAC but includes any data within the data field that has been protected for another purpose e.g. secret or private keys encrypted with the data encryption key (DEK).

Command message encryption and decryption uses the Secure Channel encryption (S-ENC) session key and the full triple DES encryption method described in appendix B.1.1.1 – *Encryption/Decryption CBC Mode*

Prior to encrypting the data, the data shall be padded. Unlike the C-MAC, this padding now becomes part of the data field and this necessitates further modification of the Lc value.

Padding of the data field to be encrypted is performed according to the following rules:

- The length of the original, “clear text”, data field is appended to the left of, and becomes part of the command data;
- If the length of the data field is now a multiple of 8, no further padding is required else continue with padding as defined in appendix B.4 - *DES Padding*.

The number of bytes appended to the data field in order to fulfill the above padding shall be added to Lc. This includes the mandatory length byte, the optional '80' and any optional binary zeroes.

The encryption can now be performed across the padded data field using the Secure Channel encryption session key (S-ENC) and the result of each encryption becomes part of the encrypted data field in the command message.

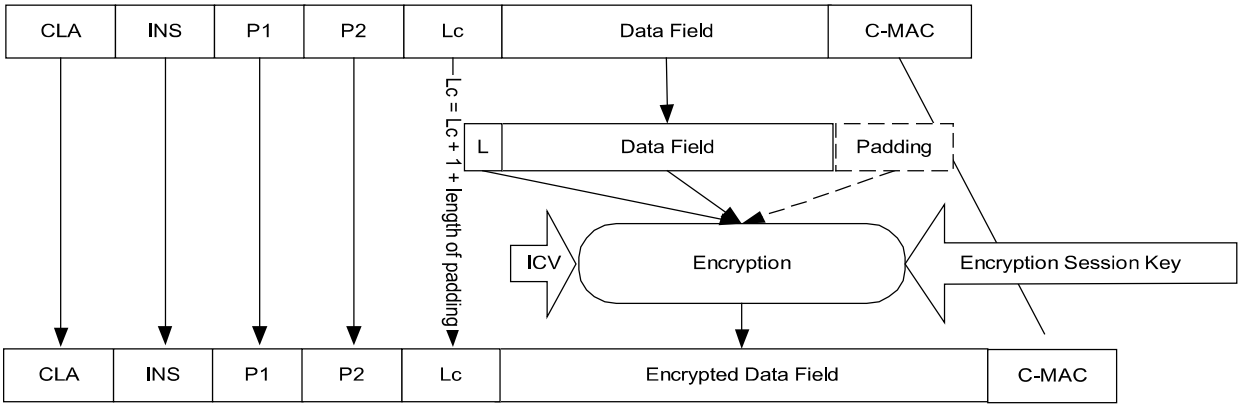


Figure D-7: APDU Data Field Encryption

Note: The ICV and the chaining are only used to link the blocks of the data currently being encrypted. For this reason the ICV for command data encryption is always binary zeroes.

The message is now prepared for transmission to the card.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0222

The card is required to first decrypt the command message and strip off any padding prior to attempting to verify the C-MAC. This decryption uses an ICV of binary zeroes and the same encryption session key employed by the off-card entity. The padding shall be removed and Lc shall be modified to reflect the length prior to encryption i.e. original clear text data plus C-MAC length.

D.3.5 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 DES keys transmitted to a card should be encrypted.

The key data encryption process uses the static data encryption key and the encryption method described in appendix B.1.1.2 - *ECB Mode*.

As all DES keys are by their very nature a multiple of 8-byte lengths no padding is required for key encryption operations. Similarly the sensitive data block length shall be constructed as a multiple of 8-byte long block before the data encryption operations: the eventual padding method is application specific.

The encryption 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; in particular, no padding is removed by the decryption operation.

D.4 Secure Channel APDU Commands

Table D-2 provides the list of minimum security requirements for SCP01 commands.

Command	Minimum Security
INITIALIZE UPDATE	None
EXTERNAL AUTHENTICATE	C-MAC

Table D-2: Minimum Security Requirements for SCP01 Commands

The following table provides the list of SCP01 command support per card Life Cycle State.

Command	OP_READY			INITIALIZED			SECURED			CARD_LOCKED		TERMINATED	
	AM SD	DM SD	SD	AM SD	DM SD	SD	AM SD	DM SD	SD	FA SD	SD	FA SD	SD
INITIALIZE UPDATE	✓	✓		✓	✓		✓	✓		✓			
EXTERNAL AUTHENTICATE	✓	✓		✓	✓		✓	✓		✓			

Table D-3: SCP01 Command Support per Card Life Cycle State

Legend:

AM SD: Security Domain with Authorized Management privilege.

DM SD: Supplementary Security Domain with Delegated Management privilege.

FA SD: Security Domain with Final Application privilege. (Note: command support for an Application with Final Application Privilege which is not a Security Domain is subject to Issuer policy, within the constraints

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0223

of what is permitted according to the card Life Cycle State.)SD: Supplementary Security Domain without Delegated Management privilege.

SD: Other Security Domain.

✓ : Support required.

Blank cell: Support optional.

Striped cell: Support prohibited.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0224

D.4.1 INITIALIZE UPDATE Command

D.4.1.1 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.

This INITIALIZE UPDATE command is not to be confused with commands of the same name in legacy applications.

D.4.1.2 Command Message

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

Code	Value	Meaning
CLA	'80' - '83	See section 11.1.4
INS	'50'	INITIALIZE UPDATE
P1	'xx'	Key Version Number
P2	'xx'	Key Identifier
Lc	'08'	Length of host challenge
Data	'xx xx…'	Host challenge
Le	'00'	

Table D-4: INITIALIZE UPDATE Command Message

D.4.1.3 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.

D.4.1.4 Reference Control Parameter P2 - Key Identifier

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.

D.4.1.5 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.

D.4.1.6 Response Message

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

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0225

Name	Length
Key diversification data	10 bytes
Key information	2 bytes
Card challenge	8 bytes
Card cryptogram	8 bytes

Table D-5: INITIALIZE UPDATE Response Message

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 and the Secure Channel Protocol identifier, here '01', used in initiating the Secure Channel Session.

The card challenge is an internally generated random number.

The card cryptogram is an authentication cryptogram.

D.4.1.7 Processing State Returned in the Response Message

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

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

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

Table D-6: INITIALIZE UPDATE Error Condition

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0226

D.4.2 EXTERNAL AUTHENTICATE Command

D.4.2.1 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.

D.4.2.2 Command Message

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

Code	Value	Meaning
CLA	'84' - '87'	See 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

Table D-7: EXTERNAL AUTHENTICATE Command Message

D.4.2.3 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 (it does not apply to this command) and within this Secure Channel Session.

b8	b7	b6	b5	b4	b3	b2	b1	Description
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.

Table D-8: EXTERNAL AUTHENTICATE Reference Control Parameter P1

D.4.2.4 Reference Control Parameter P2

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

D.4.2.5 Data Field Sent in the Command Message

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

D.4.2.6 Data Field Returned in the Response Message

The data field of the response message is not present.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0227

D.4.2.7 Processing State Returned in the Response Message

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

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

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

Table D-9: EXTERNAL AUTHENTICATE Error Condition

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0228

E Secure Channel Protocol '02'

E.1 Secure Communication

Secure Channel Protocol '02' (SCP02) supports up to 19 Supplementary Logical Channels.

E.1.1 SCP02 Secure Channel

Either the card or the off-card entity may play the role of being the secure sending and receiving entities. SCP02 provides the three followings levels of security:

Entity authentication - in which the card authenticates the off-card entity and the off-card entity may authenticate the card, proving that the off-card entity has knowledge of the same secret(s) as the card;

Integrity and Data origin authentication - in which the receiving entity (the card or off-card entity) ensures that the data being received actually came from an authenticated sending entity (respectively the off-card entity or card) in the correct sequence and has not been altered;

Confidentiality - in which data being transmitted from the sending entity (the off-card entity or card) to the receiving entity (respectively the card or off-card entity) is not viewable by an unauthorized entity.

A further level of security applies to sensitive data (e.g. secret keys) that shall always be transmitted as confidential data.

In SCP02 the "i" parameter as defined in appendix H - *GlobalPlatform Data Values and Card Recognition Data*) shall be formed as a bit map on one byte as follows:

b8	b7	b6	b5	b4	b3	b2	b1	Description
Reserved							1	3 Secure Channel Keys
Reserved							0	1 Secure Channel base key
Reserved						1		C-MAC on unmodified APDU
Reserved						0		C-MAC on modified APDU
Reserved					1			Initiation mode explicit
Reserved					0			Initiation mode implicit
Reserved				1				ICV set to MAC over AID
Reserved				0				ICV set to zero
Reserved			1					ICV encryption for C-MAC session
Reserved			0					No ICV encryption
Reserved		1						R-MAC support
Reserved		0						No R-MAC support
Reserved	1							Well-known pseudo-random algorithm (card challenge)
Reserved	0							Unspecified card challenge generation method

Table E-1: Values of Parameter "i"

Note: "i" is a subidentifier within an object identifier, and bit b8 is reserved for use in the structure of the object identifier according to ISO/IEC 8825-1.

The Security Domain shall support at least one of the following implementation options as defined by "i":

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0229

- "i" = '15': Initiation mode explicit, C-MAC on modified APDU, ICV set to zero, ICV encryption for C-MAC session, 3 Secure Channel Keys, unspecified card challenge generation method, no R-MAC;
- "i" = '1A': Initiation mode implicit, C-MAC on unmodified APDU, ICV set to MAC over AID, ICV encryption for C-MAC session, 1 Secure Channel base key. no R-MAC;
- "i" = '55': Initiation mode explicit, C-MAC on modified APDU, ICV set to zero, ICV encryption for C-MAC session, 3 Secure Channel Keys, well-known pseudo-random algorithm (card challenge), no R-MAC.

E.1.2 Entity Authentication

Off-card entity authentication is achieved through the process of initiating a Secure Channel and provides assurance to the card that it is communicating with an authenticated off-card entity. If any step in the off-card authentication process fails, the process shall be restarted (i.e. new session keys generated).

The Secure Channel initiation and off-card entity authentication implies the creation of session keys derived from card static key(s) using a Secure Channel Sequence Counter maintained by the Security Domain. The Security Domain shall manage one Sequence Counter per Secure Channel base key or Secure Channel keys sharing the same Key Version Number (see appendix E.2 - *Cryptographic Keys*).

The Sequence Counter is incremented by 1 when and only when the first C-MAC of a secure channel (started implicitly or explicitly) is verified as valid. It is incremented before the processing specific for the command. The Sequence Counter is reset to zero on creation or update of the Secure Channel key(s). When the Secure Channel key set contains more than one key (see appendix E.2 - *Cryptographic Keys*), the Sequence Counter is reset to zero on the creation or update of any one of the keys of this key set.

Note: the Sequence Counter is not updated for a C-MAC that is not the first of a Secure Channel Session or when a Secure Channel Session is started and the C-MAC is invalid. In this respect, the Sequence Counter keeps track of the number of valid Secure Channel Sessions the corresponding secure messaging key set has experienced so far. When reaching its maximum value, the Sequence Counter shall not be reset to zero. Cards are not required to support counter values beyond 32767.

E.1.2.1 Explicit Secure Channel Initiation

The Secure Channel may be 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 appendix E.5.2 - *EXTERNAL AUTHENTICATE Command*) as to what level of security is required for the current Secure Channel (integrity and/or confidentiality) and apply this level of security to all the subsequent messages exchanged between the card and the off-card entity until the end of the session. It also gives the off-card entity the possibility of selecting the Key Version Number to be used (see appendix E.5.1 - *INITIALIZE UPDATE Command*).

Note: The explicit Secure Channel Session 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 appendix E.5.1 - *INITIALIZE UPDATE Command*) by the off-card entity by passing a “host” challenge (random data unique to this session) to the card.

The card, on receipt of this challenge, generates its own “card” challenge (again random data unique to this session).

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0230

The card, using its internal Sequence Counter and static keys, creates new secret session keys and generates a first cryptographic value (card cryptogram) using one of its newly created session keys (see appendix E.4.1 - *DES Session Keys*).

This card cryptogram along with the Sequence Counter, 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 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 appendix E.5.2 - *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 cryptogram and, by performing a comparison, it is able to authenticate the off-card entity.

The off-card entity also creates a MAC to be passed back to the card and verified by the card. The verified MAC is used by the card to create the Initial Chaining Vector for the verification of the subsequent C-MAC and/or R-MAC.

When the off-card entity is successfully authenticated, the card increments its internal Secure Channel Sequence Counter.

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 section 7.3.4 - *Runtime Messaging Support*, it can be seen how an Application would use the services of a Security Domain to achieve the explicit Secure Channel initiation.

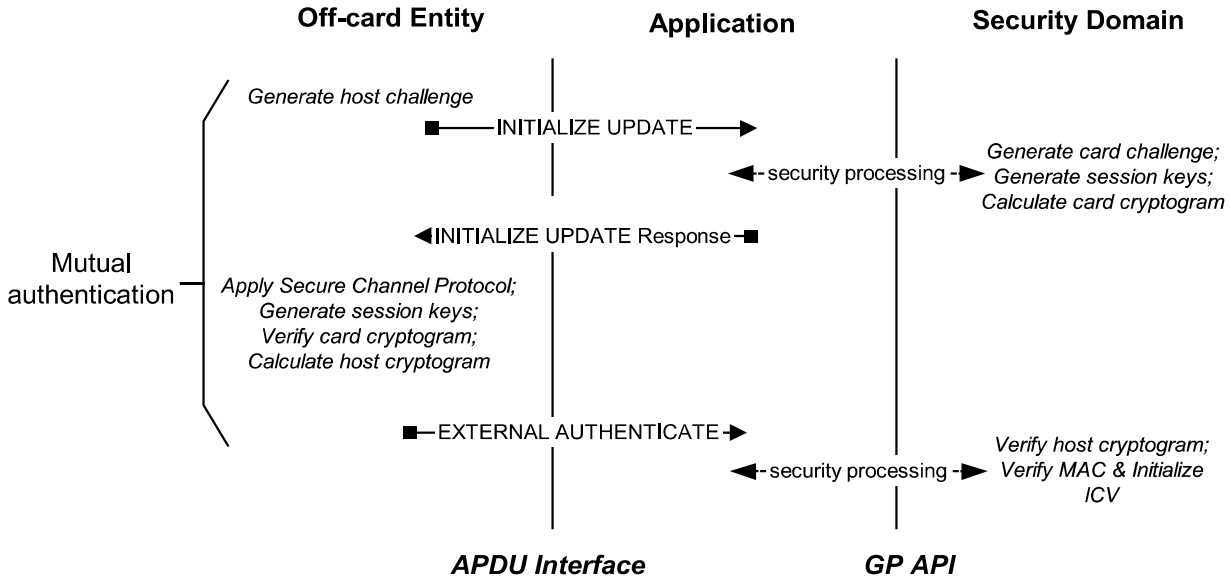


Figure E-1: Explicit Secure Channel Initiation Flow

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0231

E.1.2.2 Implicit Secure Channel Initiation

The Secure Channel is implicitly initiated when receiving the first APDU command that contains a cryptographic protection (C-MAC). The required level of security is implicitly known by both the card and off-card entity as command message integrity only. The off-card entity implicitly knows which keys are to be used and the current value of the Secure Channel Sequence Counter, or may have previously retrieved the corresponding information using a GET DATA command.

The off-card entity, using the information it knows about the card static keys and its Secure Channel Sequence Counter, creates new secret session keys and generates a C-MAC on an APDU command message.

The Secure Channel is initiated by the off-card entity by appending a C-MAC to an APDU command.

The card, on receipt of this first C-MAC, creates the C-MAC session key using its internal card static key(s) and Secure Channel Sequence Counter.

The card has the same information that the host used to generate the C-MAC. It generates the same MAC and, by performing a comparison, it authenticates the off-card entity.

When the off-card entity is successfully authenticated, the card increments its internal Secure Channel Sequence Counter.

For sensitive data decryption, the card creates the data encryption session key using its internal card static key(s) and the newly incremented value of the Secure Channel Sequence Counter. For R-MAC generation, the card creates the R-MAC session key using its internal card static key(s) and the newly incremented value of the Secure Channel Sequence Counter.

Note: In addition to command message integrity, the off-card entity may, at any moment during the Secure Channel Session, initiate response message integrity using the BEGIN R-MAC SESSION command.

E.1.3 Message Integrity

The MAC is generated by applying multiple chained DES operations (using a session key generated prior to or when opening the Secure Channel) across an APDU message.

A MAC may be generated on the following:

- C-MAC for APDU command messages (generated by the off-card entity);
- R-MAC for APDU response messages (generated by the card).

The receiving entity, on receipt of the message containing a MAC, using the same session key, performs the same operation and by comparing its generated MAC with the MAC received from the sending entity is assured of the integrity of the full command or response. (If message data confidentiality has also been applied to the message, the MAC applies to the message data field before encryption.)

The integrity of the sequence of APDU command or response messages being transmitted to the receiving entity is achieved by using the MAC from the current command or response as the (possibly encrypted) Initial Chaining Vector (ICV) for the subsequent command or response. This ensures the receiving entity that all messages in a sequence have been received. Computing the ICV is detailed in appendix E.3 - *Cryptographic Algorithms.*

E.1.4 Message Data Confidentiality

The message data field is encrypted by applying multiple chained DES operations (using a session key generated during the Secure Channel initiation process) across the entire data field of either a command

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0232

message or a response, regardless of its contents (clear text data and/or already protected sensitive data).

E.1.5 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 '02' with explicit initiation mode, the Current Security Level established in a Secure Channel Session is a bitmap combination of the following values: AUTHENTICATED, C_MAC, R_MAC, and C_DECRYPTION. 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 a successful processing of an EXTERNAL AUTHENTICATE command: AUTHENTICATED shall be cleared once the Secure Channel Session is terminated;
- C_MAC after a 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 a 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 a successful processing of an EXTERNAL AUTHENTICATE command with P1 indicating R-MAC (P1='1x'): 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_MAC after a successful processing of a BEGIN R-MAC SESSION command: R-MAC shall be cleared after a successful processing of an END R-MAC SESSION command. Note that in this case R_MAC may be combined with AUTHENTICATED, C_MAC or 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.

For Secure Channel Protocol '02' with implicit initiation mode, the Current Security Level established in a Secure Channel Session is a bitmap combination of the following values: AUTHENTICATED, C_MAC, and R_MAC. The Current Security Level shall be set as follows:

- NO_SECURITY_LEVEL when a Secure Channel Session is terminated or not yet initiated;
- AUTHENTICATED and C_MAC after a successful verification of the C- MAC of the first command message with a C-MAC. The Session Security Level shall be set to AUTHENTICATED only. C_MAC shall be cleared after the reception of the first command message without a C-MAC. AUTHENTICATED and C_MAC shall be cleared once the Secure Channel Session is terminated;
- R_MAC after a successful processing of a BEGIN R-MAC SESSION command: R-MAC shall be cleared after a successful processing of an END R-MAC SESSION command.

E.1.6 Protocol Rules

In accordance with the general rules described in chapter 10 - *Secure Communication*, the following protocol rules apply to Secure Channel Protocol '02' with explicit initiation mode:

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0233

- The successful explicit initiation of a Secure Channel Session shall set the Current Security Level to the security level value 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 not set to AUTHENTICATED (i.e. equal to NO_SECURITY_LEVEL or R-MAC only – case of a R-MAC session in progress originally initiated with a BEGIN R-MAC SESSION command):
  - 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 for applying its own security rules.
- If a Secure Channel Session is active for incoming commands (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 the Security Domain supports application data encryption and/or decryption, it shall decrypt or encrypt a block of secret data upon request. If the service is not supported or if (one of) the appropriate cryptographic key(s) is not available, the request shall be rejected but the Current Security Level, Session Security Level and Secure Channel Session in operation (if any) shall not be impacted;
- 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).

In accordance with the general rules described in chapter 10 - *Secure Communication*, the following protocol rules apply to Secure Channel Protocol '02' with implicit initiation mode:

- The successful implicit initiation of a Secure Channel Session for incoming commands shall set the Current Security Level to AUTHENTICATED and C-MAC (the R-MAC indicator remaining as is);
- The Session Security Level for incoming commands is implicit and set to AUTHENTICATED for the entire Secure Channel Session. It is reset on the normal termination or abort of a Secure Channel Session for incoming commands;

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0234

- When the Current Security Level is not set to AUTHENTICATED (i.e. equal to NO_SECURITY_LEVEL or R-MAC only), no Secure Channel Session is active for incoming commands:
  - If no secure messaging is indicated in the incoming command, no security verification of the command shall be performed. The Application processing the command is responsible to apply its own security rules;
  - If secure messaging is indicated, a new Secure Channel Session initiation for incoming commands shall be attempted and the security of the incoming command verified.
- If a Secure Channel Session is active for incoming commands (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:
  - If no secure messaging is indicated in the command, the Secure Channel Session shall remain active, the Current Security Level set to AUTHENTICATED (the R-MAC indicator remaining as is) and no further security verification of the command performed. The Application processing the command is responsible to apply its own security rules;
  - If secure messaging is indicated and the Current Security Level is set to AUTHENTICATED and C-MAC (ignoring the R-MAC indicator), the security of the incoming command shall be verified:

    If a security error is found, the command shall be rejected with a security error, the Secure Channel Session aborted and Current Security Level reset to NO_SECURITY_LEVEL or R-MAC only (in case of a R-MAC session in progress);

    Otherwise, the Secure Channel Session shall remain active and the Current Security Level unmodified. The Application is responsible for further processing the command.
  - If secure messaging is indicated and the Current Security Level is set to AUTHENTICATED (ignoring the R-MAC indicator), the current Secure Channel Session for incoming commands shall be terminated and the Current Security Level reset to NO_SECURITY_LEVEL or R-MAC only (in case of a R-MAC session in progress). A new Secure Channel Session initiation for incoming commands shall be attempted and the security of the incoming command verified.
- 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:
  - 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).

E.2 Cryptographic Keys

A Security Domain, including the Issuer Security Domain, shall have at least one key set to be used in the initiation and use of a Secure Channel. This key set shall contain at least one double length DES key, the Secure Channel base key. This Secure Channel base key is only used to generate session keys during the initiation of a Secure Channel.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0235

Key	Usage	Length	Remark
Secure Channel base key	Secure Channel authentication, MAC verification, & sensitive data decryption (DES)	16 bytes	Mandatory

Table E-2: SCP02 - Security Domain Secure Channel Base Key

A Security Domain's, including the Issuer Security Domain's, Secure Channel key set may contain 3 double length DES keys: the Secure Channel encryption key (S-ENC), the Secure Channel MAC key (S-MAC), and the data encryption key (DEK) for decrypting sensitive data, e.g. secret or private keys. These keys are only used to generate session keys during the initiation of a Secure Channel.

Key	Usage	Length	Remark
Secure Channel encryption key (S-ENC)	Secure Channel authentication & encryption (DES)	16 bytes	Mandatory
Secure Channel message authentication code key (S-MAC)	Secure Channel MAC verification and generation (DES)	16 bytes	Mandatory
Data encryption key (DEK)	Sensitive data decryption (DES)	16 bytes	Mandatory

Table E-3: SCP02 - Security Domain Secure Channel Keys

E.3 Cryptographic Algorithms

The cryptographic and hashing algorithms described in appendix B - *Algorithms (Cryptographic and Hashing)* apply to SCP02. This section defines the additional requirements for SCP02.

E.3.1 Cipher Block Chaining (CBC)

The Initial Chaining Vector (ICV) used for chained data encryption in CBC mode is always 8 bytes of binary zero ('00').

The Initial Chaining Vector (ICV) used for chained message integrity in CBC mode is always 8 bytes and initialized during the Secure Channel initiation.

E.3.2 Message Integrity ICV using Explicit Secure Channel Initiation

When using explicit Secure Channel initiation, SCP02 mandates the use of a MAC on the EXTERNAL AUTHENTICATE command.

For the EXTERNAL AUTHENTICATE command MAC verification, the ICV is set to zero.

Once successfully verified, the MAC of the EXTERNAL AUTHENTICATE command becomes the ICV for the subsequent C-MAC verification and/or R-MAC generation.

E.3.3 Message Integrity ICV using Implicit Secure Channel Initiation

When using implicit Secure Channel Session initiation, the ICV shall be a MAC computed on the AID of the selected Application. The ICV for the first C-MAC calculation of a new Secure Channel Session is calculated as follows:

- Apply reversible padding to the AID of the selected Application as defined in appendix B.4 - *DES Padding*;
- Calculate a MAC as defined in appendix B.1.2.2 - *Single DES Plus Final Triple DES* with the C-MAC key over the padded Application AID with an ICV value of binary zeroes.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0236

The resulting MAC is the ICV for the first C-MAC of the Secure Channel Session. This ICV makes the Secure Channel Session application specific.

The ICV for the first R-MAC of a Secure Channel Session is calculated the same way, except that the R-MAC key is used in step 2 for the MAC calculation.

E.3.4 ICV Encryption

As an enhancement to the C-MAC mechanism, the ICV is encrypted before being applied to the calculation of the next C-MAC. The encryption mechanism used is single DES with the first half of the Secure Channel C-MAC session key. The first ICV of a session is not encrypted.

E.4 Cryptographic Usage

E.4.1 DES Session Keys

DES session keys are generated every time a Secure Channel is initiated. These session keys may be used for subsequent commands if secure messaging is required.

Session keys are generated to ensure that a different set of keys is used for each secure communication session.

DES session keys are created using the static Secure Channel key(s), a constant, the current value of the Secure Channel Sequence Counter, and a padding of binary zeroes. Creating session keys is performed as follows:

- Generating the Secure Channel C-MAC session keys using the Secure Channel base key or MAC key (S-MAC) and the session keys derivation data with a constant of '0101';
- Generating the Secure Channel R-MAC session keys using the Secure Channel base key or MAC key (S-MAC) and the session keys derivation data with a constant of '0102';
- Generating the Secure Channel encryption session keys using the Secure Channel base key or encryption key (S-ENC) and the session keys derivation data with a constant of '0182';
- Generating the Secure Channel data encryption session keys using the Secure Channel base key or data encryption key (DEK) and the session keys derivation data with a constant of '0181'.

The DES operation used to generate these keys is always triple DES in CBC mode.

R-MAC session keys are generated using the current value of the sequence counter at the time the R-MAC session is initiated. If the R-MAC session starts with the EXTERNAL AUTHENTICATE command, the same sequence counter value is used to generate the R-MAC session key as is used to generate the other session keys for the same Secure Channel Session.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0237

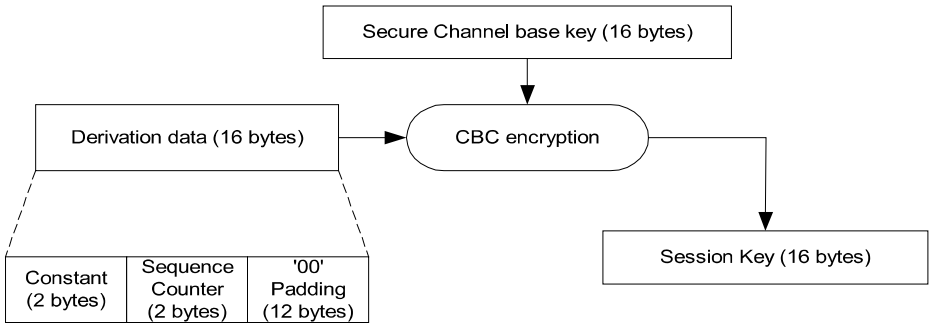


Figure E-2: Create Secure Channel Session Key from the Base Key

E.4.2 Authentication Cryptograms in Explicit Secure Channel Initiation

Both the card and the off-card entity (host) generate an authentication cryptogram. The off-card entity verifies the card cryptogram and the card verifies the host cryptogram. Generating or verifying an authentication cryptogram uses the S-ENC session key and the signing method described in appendix B.1.2.1 - *Full Triple DES*.

E.4.2.1 Card Authentication Cryptogram

The generation and verification of the card cryptogram is performed by concatenating the 8-byte host challenge, 2-byte Sequence Counter, and 6-byte card challenge resulting in a 16-byte block.

Applying the same padding rules defined in appendix B.4 - *DES Padding*, the data shall be padded with a further 8-byte block ('80 00 00 00 00 00 00 00').

The signature method, using the S-ENC session key and an ICV of binary zeroes, is applied across this 24-byte block and the resulting 8-byte signature is the card cryptogram.

E.4.2.2 Host Authentication Cryptogram

The generation and verification of the host cryptogram is performed by concatenating the 2-byte Sequence Counter, 6-byte card challenge, and 8-byte host challenge resulting in a 16-byte block.

Applying the same padding rules defined in appendix B.4 - *DES Padding*, the data shall be padded with a further 8-byte block ('80 00 00 00 00 00 00 00').

The signature method, using the S-ENC session key and an ICV of binary zeroes, is applied across this 24-byte block and the resulting 8-byte signature is the host cryptogram.

E.4.2.3 Card Challenge

The card challenge is either a random or pseudo-random number that shall be unique to a Secure Channel Session. A pseudo-random card challenge may be generated as follows:

- The AID of the Application requesting to open the Secure Channel is padded according to the padding rules defined in appendix B.4 - *DES Padding*;
- A MAC is calculated across the padded data as defined in appendix B.1.2.2 - Single DES Plus Final Triple DES MAC, using the C-MAC session key and an ICV of binary zeroes;

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0238

- The six leftmost bytes of the resultant MAC constitute the card challenge.

E.4.3 Authentication Cryptogram in Implicit Secure Channel Initiation

When using the implicit Secure Channel Session initiation, the authentication cryptogram is the first C-MAC received by the card from the off-card entity.

The rules for C-MAC generation and verification are detailed in the following subsection.

E.4.4 APDU Command C-MAC Generation and Verification

A C-MAC is generated by an off-card entity and applied across the full APDU command being transmitted to the card including the header and the data field in the command message. It does not include Le.

C-MAC generation and verification uses the Secure Channel C-MAC session key, an ICV and the signature method described in appendix B.1.2.2 - *Single DES Plus Final Triple DES*. (Prior to using the ICV, the ICV can be encrypted as described in appendix E.3.4 - *ICV Encryption*)

The ICV is used to chain the commands for command sequence integrity; the initial value of the ICV is described in appendix E.3 - *Cryptographic Algorithms*. For any subsequent command following the first successful C-MAC verification, the ICV is the C-MAC value successfully verified for the previous command received by the card.

The signature method, using the Secure Channel C-MAC session key and the ICV, is applied across the padded command message and the resulting 8-byte signature is the C-MAC. The rules for C-MAC padding are as defined in appendix B.4 - *DES Padding.*

To reflect the presence of a C-MAC in the command message, the command header shall be modified 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 to indicate that this APDU command includes secure messaging. This is achieved by either setting to '1' bit 3 of a class byte indicating a logical channel number 0 to 4 (unprotected CLA set to '00' - '03' or '80' - '83') or by setting to '1' bit 6 of a class byte indicating a logical channel number 4 to 19 (unprotected CLA set to '40' - '4F' or 'C0' - 'CF'): see section 11.1.4. The C-MAC generation and verification does not reflect the logical channel information included in the class byte of the command: the logical channel number is always assumed to be zero in the generation or verification of the C-MAC.

The C-MAC is appended at the end of the APDU command message excluding any padding but including the modifications made to the command header (class and Lc).

The two following methods are defined for the C-MAC generation:

- C-MAC generation on unmodified APDU: padding of the APDU command is required prior to the C-MAC generation being performed, and modification of the APDU command header is required after the C-MAC generation has been performed;

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0239

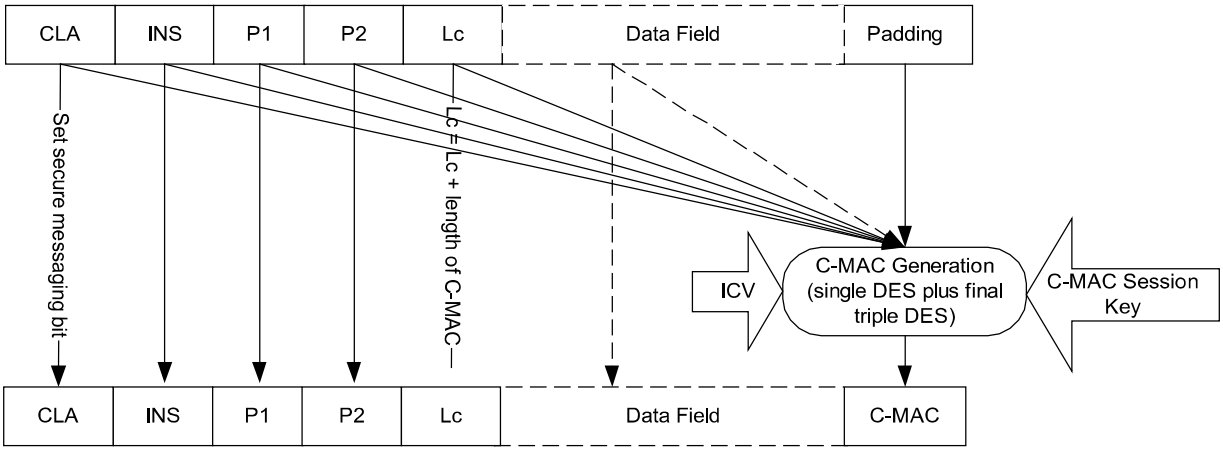


Figure E-3: C-MAC Generation on Unmodified APDU

- C-MAC generation on modified APDU: padding of the APDU command and modification of the command header is required prior to the C-MAC generation being performed.

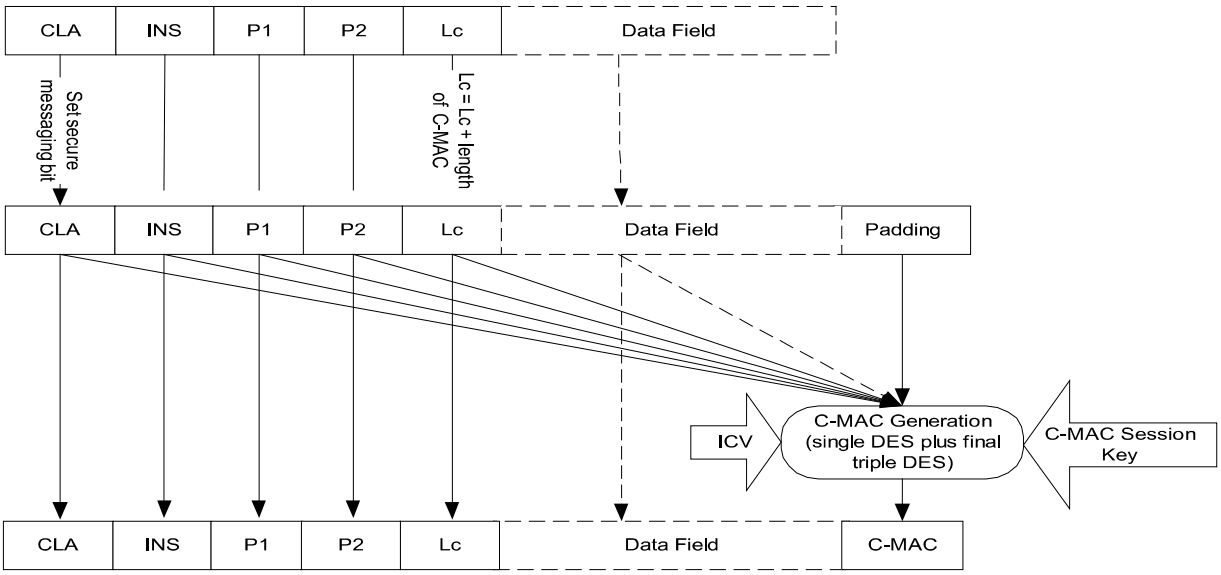


Figure E-4: C-MAC Generation on Modified APDU

If no other secure messaging is required, the message is now prepared for transmission to the card.

The card, in order to verify the C-MAC, shall perform the same padding mechanism to the data and use the same ICV and C-MAC session key employed by the off-card entity in order to verify the C-MAC. The verified C-MAC shall be retained and used as the ICV for any subsequent C-MAC verification. This is true regardless of whether the APDU completed successfully or not i.e. a verified C-MAC shall never be discarded in favor of a previous verified C-MAC value.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0240

If the Secure Channel Session is occurring on a Supplementary Logical Channel, the logical channel number is added to the class byte of the message after the C-MAC has been generated and prior to the message being transmitted to the card. The card shall remove any logical channel information from the class byte (i.e. assume the logical channel number is zero) prior to verifying the C-MAC.

E.4.5 APDU Response R-MAC Generation and Verification

When using explicit Secure Channel Session initiation, the off-card entity instructs the card whether R-MAC generation applies to all the subsequent command/response messages. The APDU response message integrity does not include the EXTERNAL AUTHENTICATE command/response pair and lasts until the end of the Secure Channel Session.

When using implicit Secure Channel Session initiation, the off-card entity instructs the card to start APDU response message integrity using the BEGIN R-MAC SESSION command (see appendix E.5.3 - *BEGIN R-MAC SESSION Command*). The APDU response message integrity includes the BEGIN R-MAC SESSION command/response pair and lasts until reception of the END R-MAC SESSION command (see appendix E.5.4 - *END R-MAC SESSION Command*).

The R-MAC is computed on the following data block:

- The stripped APDU command message, i.e. without any C-MAC and modified command header (the logical channel number is always assumed to be zero). In the case of a case 1 or case 2 command, Lc is always present and set to zero;
- In case of a successful execution or a warning, the response data preceded with a byte Li that codes its length modulo 256. Li is generated by the Security Domain. If there is no response data this byte is present and is set to zero;
- In case of an error, a byte set to '00' indicating no response data;
- The status bytes.

When an R-MAC is returned in response to every command, in the event of case 1 and case 3 commands received by the card, these commands shall be processed respectively as case 2 and case 4 commands with Le set to zero.

The signature method, using the Secure Channel R-MAC session key and the ICV, is applied across the padded data block and the resulting 8-byte signature is the R-MAC. The rules for R-MAC padding are as defined in appendix B.4 - *DES Padding.*

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0241

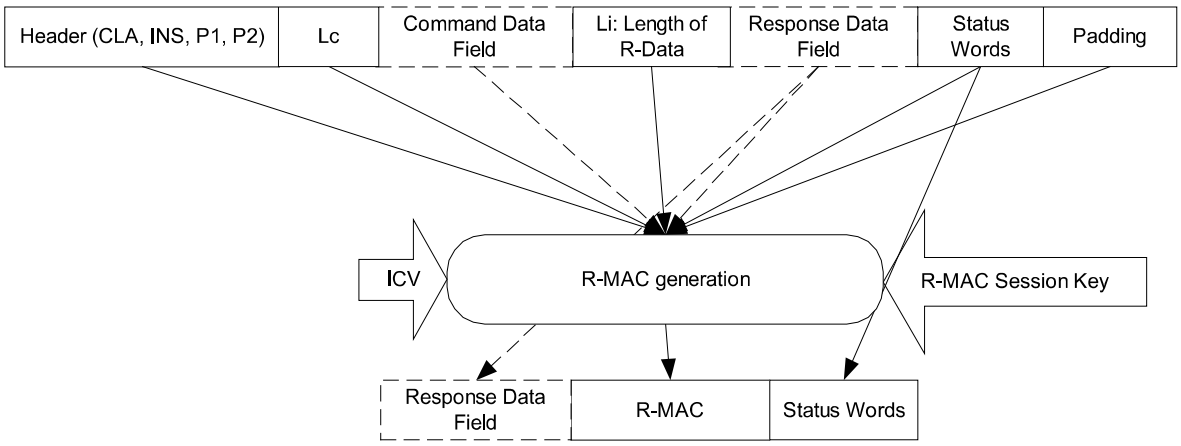


Figure E-5: R-MAC Generation

The R-MAC can be retrieved from the card by sending an END R-MAC SESSION command. The END R-MAC SESSION command allows the off-card entity to instruct the card to either terminate or continue the R-MAC session. The END R-MAC SESSION command is not included in the R-MAC computation.

The off-card entity, in order to verify the R-MAC, shall perform the same padding mechanism to the command/response pair and use the same ICV and R-MAC session key employed by the card in order to verify the R-MAC. The generated R-MAC shall be retained and used as the ICV for any subsequent R-MAC generation. (This is true regardless of whether the APDU command completed successfully or not.)

If the Secure Channel Session is occurring on a Supplementary Logical Channel, the logical channel information is not included in any of the R-MAC computations. The R-MAC is generated with a logical channel number assumed to be zero and the off-card entity is either not aware of, or removes, any logical channel information from the class byte of the corresponding command when verifying the R-MAC.

E.4.6 APDU Command Data Field Encryption and Decryption

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

Note that, with an implicit initiation of the Secure Channel, command message data field encryption does not apply.

If confidentiality is required, the off-card entity encrypts the data field of the command message being transmitted to the card. This excludes the header and the C-MAC but includes any data within the data field that has been protected for another purpose e.g. secret or private keys encrypted with the data encryption session key.

Command message encryption and decryption uses the Secure Channel encryption session key and the encryption method described in appendix B.1.1.1 – *Encryption/Decryption CBC Mode*.

Prior to encrypting the data, the data shall be padded. Unlike the C-MAC, this padding now becomes part of the data field and this necessitates further modification of the Lc value.

Padding of the data field to be encrypted is performed according to appendix B.4 - *DES Padding*.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0242

The number of bytes appended to the data field in order to fulfill the above padding shall be added to Lc. This includes the mandatory '80' and any optional binary zeroes.

The encryption can now be performed across the padded data field using the Secure Channel encryption session key and the result of each encryption becomes part of the encrypted data field in the command message.

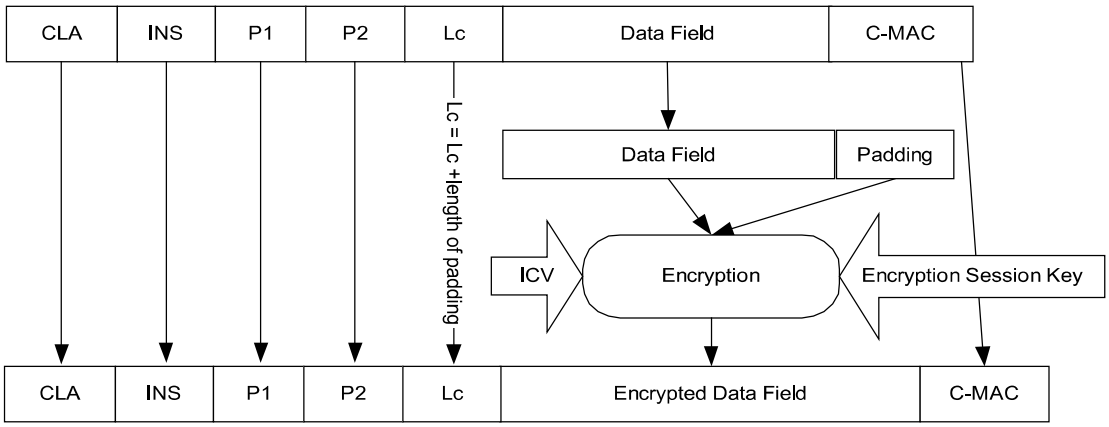


Figure E-6: APDU Command Data Field Encryption

Note: The ICV and the chaining are only used to link the blocks of the data currently being encrypted. For this reason the ICV for command data encryption is always binary zeroes.

The message is now prepared for transmission to the card. The card is required to first decrypt the command message and strip off any padding prior to attempting to verify the C-MAC. This decryption uses an ICV of binary zeroes and the same encryption session key employed by the off-card entity. The padding shall be removed and Lc shall be modified to reflect the length prior to encryption i.e. original clear text data plus C-MAC length.

E.4.7 Sensitive Data Encryption and Decryption

Data encryption is used when transmitting sensitive data to the card and is over and beyond the Current Security Level required for the Secure Channel; For instance all DES keys transmitted to a card (e.g. in a PUT KEY command) should be encrypted.

The data encryption process uses the data encryption session key and the encryption method described in appendix B.1.1.2 – *Encryption/Decryption ECB Mode* when using explicit initiation of the Secure Channel and appendix B.1.1.1 – *Encryption/Decryption CBC Mode* when using implicit initiation of the Secure Channel.

As all DES keys are by their very nature a multiple of 8-byte lengths no padding is required for key encryption operations. Similarly the sensitive data block length shall be constructed as a multiple of 8-byte long block before the data encryption operations: the eventual padding method is application specific.

The encryption is performed across the sensitive data and the result of each encryption becomes part of the encrypted data. This encrypted 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: in particular, no padding is removed by the decryption operation.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0243

E.5 Secure Channel APDU Commands

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

Table E-4: SCP02 Command Support

Table E-5 summarizes the minimum security requirements for the APDU commands.

Command	Minimum Security	
	Explicit	Implicit
INITIALIZE UPDATE	None	Dependent on the Issuer security policy
EXTERNAL AUTHENTICATE	C-MAC	
BEGIN R-MAC SESSION	Secure Channel initiation	
END R-MAC SESSION	Secure Channel initiation	

Table E-5: Minimum Security Requirements for SCP02 Commands

Note that Table E-4 shall be used in conjunction with Table E-6 to determine SCP02 command support requirements. The following table provides the list of SCP02 command support per card Life Cycle State.

Command	OP_READY			INITIALIZED			SECURED			CARD_LOCKED		TERMINATED	
	AM SD	DM SD	SD	AM SD	DM SD	SD	AM SD	DM SD	SD	FA SD	SD	FA SD	SD
INITIALIZE UPDATE	✓	✓		✓	✓		✓	✓		✓			
EXTERNAL AUTHENTICATE	✓	✓		✓	✓		✓	✓		✓			
BEGIN R-MAC SESSION													
END R-MAC SESSION													

Table E-6: SCP02 Command Support per card Life Cycle State

Legend of Table E-4 and Table E-6*:*

AM SD: Security Domain with Authorized Management privilege.

DM SD: Supplementary Security Domain with Delegated Management privilege.

FA SD: Security Domain with Final Application privilege. (Note: command support for an Application with Final Application Privilege which is not a Security Domain is subject to Issuer policy, within the constraints of what is permitted according to the card Life Cycle State.)

SD: Other Security Domain.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0244

✓ : Support required.

Blank cell: Support optional.

Striped cell: Support prohibited.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0245

E.5.1 INITIALIZE UPDATE Command

E.5.1.1 Definition and Scope

The INITIALIZE UPDATE command is used, during explicit initiation of a Secure Channel, to transmit card and 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.

This INITIALIZE UPDATE command is not to be confused with commands of the same name in legacy applications.

E.5.1.2 Command Message

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

Code	Value	Meaning
CLA	'80' - '83' or 'C0' - 'CF'	See section 11.1.4
INS	'50'	INITIALIZE UPDATE
P1	'xx'	Key Version Number
P2	'00'	Reference control parameter P2
Lc	'08'	Length of host challenge
Data	'xx xx…'	Host challenge
Le	'00'	

Table E-7: INITIALIZE UPDATE Command Message

E.5.1.3 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.

E.5.1.4 Reference Control Parameter P2

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

E.5.1.5 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.

E.5.1.6 Response Message

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

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0246

Name	Length
Key diversification data	10 bytes
Key information	2 bytes
Sequence Counter	2 bytes
Card challenge	6 bytes
Card cryptogram	8 bytes

Table E-8: INITIALIZE UPDATE Response Message

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 and the Secure Channel Protocol identifier, here '02', used in initiating the Secure Channel Session.

The Sequence Counter is an internal incremental counter used for creating session keys.

The card challenge is an internally generated random number.

The card cryptogram is an authentication cryptogram.

E.5.1.7 Processing State Returned in the Response Message

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

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

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

Table E-9: INITIALIZE UPDATE Error Condition

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0247

E.5.2 EXTERNAL AUTHENTICATE Command

E.5.2.1 Definition and Scope

The EXTERNAL AUTHENTICATE command is used by the card, during explicit initiation of a Secure Channel, 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.

E.5.2.2 Command Message

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

Code	Value	Meaning
CLA	'84' - '87' or 'E0' - 'EF'	See 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

Table E-10: EXTERNAL AUTHENTICATE Command Message

E.5.2.3 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 (it does not apply to this command) and within this Secure Channel Session.

b8	b7	b6	b5	b4	b3	b2	b1	Description
0	0	1	1	0	0	1	1	RFU
0	0	1	1	0	0	0	1	RFU
0	0	1	1	0	0	0	0	RFU
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	1	0	0	0	0	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.

Table E-11: EXTERNAL AUTHENTICATE Reference Control Parameter P1

E.5.2.4 Reference Control Parameter P2

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

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0248

E.5.2.5 Data Field Sent in the Command Message

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

E.5.2.6 Data Field Returned in the Response Message

The data field of the response message is not present.

E.5.2.7 Processing State Returned in the Response Message

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

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

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

Table E-12: EXTERNAL AUTHENTICATE warning Code

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0249

E.5.3 BEGIN R-MAC SESSION Command

E.5.3.1 Definition and Scope

The BEGIN R-MAC SESSION command is used to initiate a Secure Channel Session for APDU response message integrity. At any time, the BEGIN R-MAC SESSION command may be issued to the card in order to initiate a R-MAC session.

E.5.3.2 Command Message

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

Code	Value	Meaning
CLA	'80' - '87', 'C0' -'CF' or 'E0' - 'EF'	See 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

Table E-13: BEGIN R-MAC SESSION Command Message

E.5.3.3 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).

b8	b7	b6	b5	b4	b3	b2	b1	Description
0	0	1	1	0	0	0	0	Response Encryption and R-MAC (RFU)
0	0	0	1	0	0	0	0	R-MAC
0	0	0	0	0	0	0	0	No secure messaging expected

Table E-14: BEGIN R-MAC SESSION Reference Control Parameter P1

When P1 is set to '10' each APDU response message during the R-MAC session includes a R-MAC.

When P1 is set to '00' only the END R-MAC SESSION response message will contain a R-MAC.

E.5.3.4 Reference Control Parameter P2

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

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

Table E-15: BEGIN R-MAC SESSION Reference Control Parameter P2

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0250

E.5.3.5 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.

The following table details the BEGIN R-MAC SESSION data field:

Length	Name	Presence
1	Length of data	Mandatory
0-24	data	Conditional

Table E-16: BEGIN R-MAC SESSION Command Data Field

E.5.3.6 Data Field Returned in the Response Message

The data field of the response message is not present.

E.5.3.7 Processing State Returned in the Response Message

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

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

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

Table E-17: BEGIN R-MAC SESSION Error Conditions

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0251

E.5.4 END R-MAC SESSION Command

E.5.4.1 Definition and Scope

The END R-MAC SESSION command is used to terminate a Secure Channel Session for APDU response message integrity or to retrieve the current R-MAC without ending the R-MAC Session. The END R-MAC SESSION command may be issued to the card at any time during an R-MAC session, .

E.5.4.2 Command Message

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

Code	Value	Meaning
CLA	'80' - '87', 'C0' -'CF' or 'E0' - 'EF'	See section 11.1.4
INS	'78'	END R-MAC SESSION
P1	'00'	Reference control parameter P1
P2	'01' or '03'	Reference control parameter P2
Lc	'xx'	Length of data field, if any
Data	'xx xx…'	C-MAC, if needed
Le	'00'	

Table E-18: END R-MAC SESSION Command Message

E.5.4.3 Reference Control Parameter P1

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

E.5.4.4 Reference Control Parameter P2

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

b8	b7	b6	B5	b4	b3	b2	b1	Description
0	0	0	0	0	0	1	1	End R-MAC session & return R-MAC
0	0	0	0	0	0	0	1	Return R-MAC

Table E-19: END R-MAC SESSION Reference Control Parameter P2

E.5.4.5 Data Field Sent in the Command Message

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

E.5.4.6 Data Field Returned in the Response Message

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

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0252

E.5.4.7 Processing State Returned in the Response Message

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

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

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

Table E-20: END R-MAC SESSION Error Conditions

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0253

F Secure Channel Protocol '10'

F.1 Secure Communication

Secure Channel Protocol '10' (SCP10) supports up to 19 Supplementary Logical Channels.

F.1.1 SCP10 Secure Channel

SCP10 is a Secure Channel Protocol based on asymmetric cryptography and Public Key infrastructure (PKI). The Secure Channel Protocol operates between a Security Domain on the card (or the Issuer Security Domain) and an Off-Card Entity (OCE) which may be the Application Provider (or Card Issuer) or another party.

F.1.1.1 Synchronous Mode

The synchronous implementation of SCP10 provides the following levels of security: Authentication - in which the Security Domain authenticates the Off-Card Entity and the Off-Card Entity may authenticate the Security Domain. There are two aspects to this: Key Authentication which involves establishing mutual trust in each other's public key (PK); and Entity Authentication which involves establishing that the other party in the Secure Channel Session is the authentic owner of that public key.

Integrity and data origin authentication - in which the Security Domain and the Off-Card Entity ensure that the data being received from the other entity actually came from its claimed source in the correct sequence and has not been altered.

Confidentiality - in which confidential data is not viewable by an unauthorized entity.

A further level of security applies to specific sensitive data (e.g. cryptographic keys) which may be encrypted using a mechanism that is not part of the Secure Channel Session security.

F.1.1.2 Asynchronous Mode

The asynchronous mode allows off-line pre-packaging of encrypted and signed content and provides the following levels of security:

Integrity and data origin authentication – in which the Security Domain confirms the public-key-signed signature of the supplied content. This enables that the content being received from the other entity actually came from its claimed source in the correct sequence and has not been altered.

Confidentiality – in which confidential data is not viewable by an unauthorized entity. Confidential content is encrypted with the public key of the desired Security Domain.

F.1.2 Initiating a Secure Channel

The steps involved in initiating a Secure Channel Protocol are as follows:

1. **Certificate Verification (optional) -** the Security Domain and the Off-Card Entity (OCE) may need to traverse and verify a chain of certificates from established trust points down to each other's public key. The extent to which this is needed depends on what keys are currently validated by each party: the null condition is where they have both already validated each other's public key, in which case no explicit Certificate Verification is necessary;

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0254

2. **Entity Authentication -** the Security Domain and optionally the Off-Card Entity further check the authenticity of the other party by verifying the signature of a challenge sent to the other party;
3. **Session key establishment -** the two parties establish symmetric session keys for subsequent secure messaging.

Only explicit Secure Channel initiation is supported in SCP10. There is no specific command to terminate a session.

Some of the variations on the Secure Channel Protocol supported by the card or the Security Domain are announced in the parameter "i" in Card Recognition Data or Security Domain Management Data. See appendix H - *GlobalPlatform Data Values and Card Recognition Data* for details of Card Recognition Data and Security Domain Management Data. The value "i" is coded on one byte as a bit map as follows:

b8	b7	b6	b5	b4	b3	b2	b1	Description
Not available	0	0	0	0	0	-	0	Key Transport
Not available	0	0	0	0	0	-	1	Key Agreement
Not available	0	0	0	0	0	0	-	Signature with message recovery
Not available	0	0	0	0	0	1	-	Signature without message recovery

Table F-1: Values of Parameter "i"

Note: "i" is a subidentifier within an object identifier, and bit b8 is reserved for use in the structure of the object identifier according to ISO/IEC 8825-1.

Key transport and key agreement relate to the process of establishing session keys for the Secure Channel Session.

- With **key agreement** the Security Domain and the Off-Card Entity exchange secret values when the Secure Channel is being initiated, and session keys are then derived from those secrets using an algorithm known to both the Off-Card Entity and the Security Domain;
- With **key transport** the Security Domain receives session keys to be used for the Secure Channel Session from the Off-Card Entity during Secure Channel initiation.

Signature with message recovery and signature without message recovery refer to the signature scheme used for digital signatures in data messages during Entity Authentication. (Note that these mechanisms apply also with the signatures on digital certificates, but the value "i" does not refer to this.)

- With **signature with message recovery**, part or all of the message data that is signed is contained in the signature block and is recovered during the process of verifying the signature. Signature with message recovery is standardized in ISO 9796-2;
- With **signature without message recovery**, the signature does not contain any part of the message data that is signed, but comprises an appendix to the complete message. Such a scheme is also known as a signature scheme 'with appendix'. Signature without message recovery is standardized in PKCS#1, and is also used in X.509.

The Security Domain may support any combination of values of parameter "i", but shall support at least one of the following options as defined by "i":

- "i" = '01': Signature with recovery, key agreement;
- "i" = '02': Signature without recovery, key transport.

There is no requirement for a Security Domain to support more than one certificate format.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0255

F.1.3 Certificate Verification

F.1.3.1 Overview

The Security Domain verifies a certificate of the Off-Card Entity to establish the validity of its public key. A certificate must be issued by the Trust Point for External Authentication (TP_EX) or a subordinate key authority of the TP_EX. On the other hand, the Off-Card Entity establishes the validity of the Security Domain's public key by verifying a certificate issued by the Key Authority of the Security Domain. The Security Domain shall hold a single (default) public key of the Trust Point for External Authentication (PK.TP_EX.AUT) and may also hold the validated public keys of other authorities in a certificate chain.

The Off-Card Entity may know implicitly what other public keys have already been validated by the Security Domain, and can use this knowledge to reduce the number of certificates the Security Domain is asked to verify, potentially down to zero.

By default the Security Domain expects the first certificate presented for verification to be signed by the PK.TP_EX.AUT, and each subsequent certificate to be signed by the public key validated with the previous certificate. The Off-Card Entity may request the Security Domain to use a different default initial public key by indicating it with a MANAGE SECURITY ENVIRONMENT command.

The Off-Card Entity establishes the validity of the Security Domain's public key by checking a (chain of) certificate(s).

The minimum set of keys and certificates to be stored by a Security Domain that supports asymmetric Secure Channel Protocol '10' shall be as follows:

- One Public Key for Trust Point for External Authentication (PK.TP_EX.AUT);
- One Security Domain Private Key (SK.SD.AUT);
- One Security Domain Public Key (PK.SD.AUT); and
- One Security Domain Certificate (CERT.SD.AUT) corresponding to the Security Domain Public Key and signed by the Security Domain Key Authority.

The Security Domain may also hold:

- The Off-Card Entity's Public Key(s) (PK.OCE.AUT) for External Authentication which has been validated;
- The Key Authority's Public Key(s) (PK.KA_EX.AUT) for External Authentication in a valid certificate chain from the Trust Point for External Authentication to the Off-Card Entity; and
- The Key Authority's Certificate(s) for Internal Authentication (CERT.KA_IN.AUT) in a valid certificate chain from the Trust Point for Internal Authentication to the Security Domain.

The minimum set of keys and certificates to be available to an Off-Card Entity (OCE) that wishes to communicate with a specific Security Domain is assumed to be as follows:

- One Public Key for Trust Point for Internal Authentication (PK.TP_IN.AUT));
- One Off-Card Entity Public Key (PK.OCE.AUT);
- One Off-Card Entity Certificate (CERT.OCE.AUT) corresponding to the Off-Card Entity Public Key and signed by the Off-Card Entity's Key Authority;
- The Off-Card Entity Trust Point Certification Public Key (PK.TP_OCE.AUT) that participates in a chain of certificates down to the Security Domain's Public Key.

The Off-Card Entity may also hold:

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0256

- The Key Authority's Certificate(s) for External Authentication (CERT.KA_EX.AUT) in a valid certificate chain from the Trust Point for External Authentication to the Off-Card Entity.

Note that the Security Domain's owner may be considered to be the Application Provider (for a Security Domain) or the Card Issuer (for the Issuer Security Domain).

The first example in Figure F-1 is a simple case where the Trust Point for External Authentication (TP_EX) and the Trust Point for Internal Authentication (TP_IN) certify the public key of the Key Authority of the Off-Card Entity and the Security Domain respectively, and the Key Authority of the Off-Card Entity (KA_OCE) certifies the public key of the Off-Card Entity.

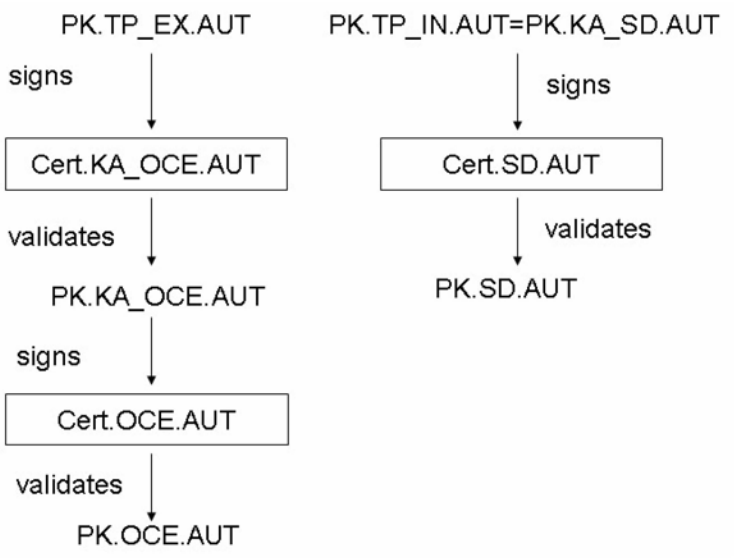


Figure F-1 - Certificate Chains - Example a

The second example in Figure F-2 illustrates that the Trust Point for External Authentication (TP_EX) and Internal Authentication (TP_IN) directly certifies the public key of both the Off-Card Entity and the Security Domain's Key Authority; and the Security Domain's Key Authority in turn certifies the Security Domain's public key.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0257

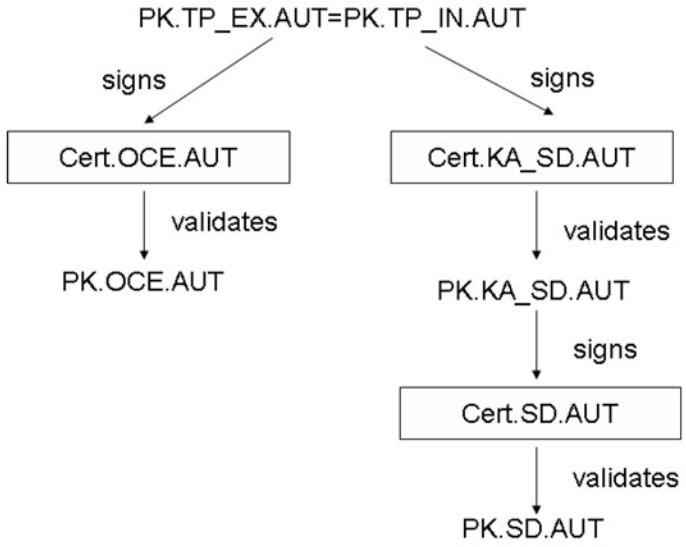


Figure F-2 - Certificate Chains - Example b

The third example in Figure F-3 illustrates that the Trust Point for External Authentication (TP_EX) and Internal Authentication (TP_IN) cross-certify each other's public keys, and there are two certificate chains down to the Security Domain and OCE public keys.

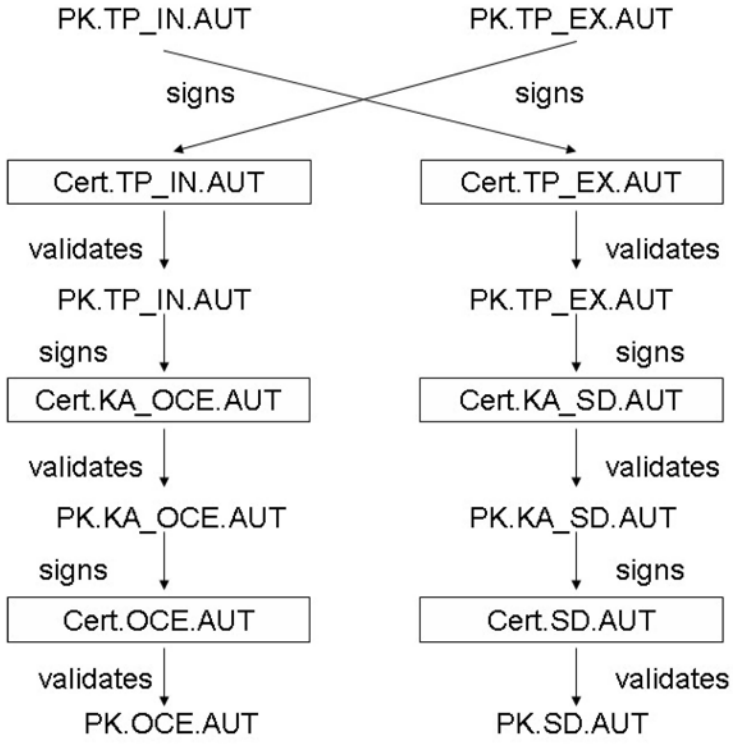


Figure F-3 Certificate Chains - Example c

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0258

F.1.3.2 Certificate Verification Process Flow

The following flow is an example of the Certificate Verification stage of Secure Channel initiation. Expanding on the authentication processing shown in the flow described in section 7.3.4 - *Runtime Messaging Support* it can be seen how an Application would use the services of a Security Domain to perform Certificate Verification.

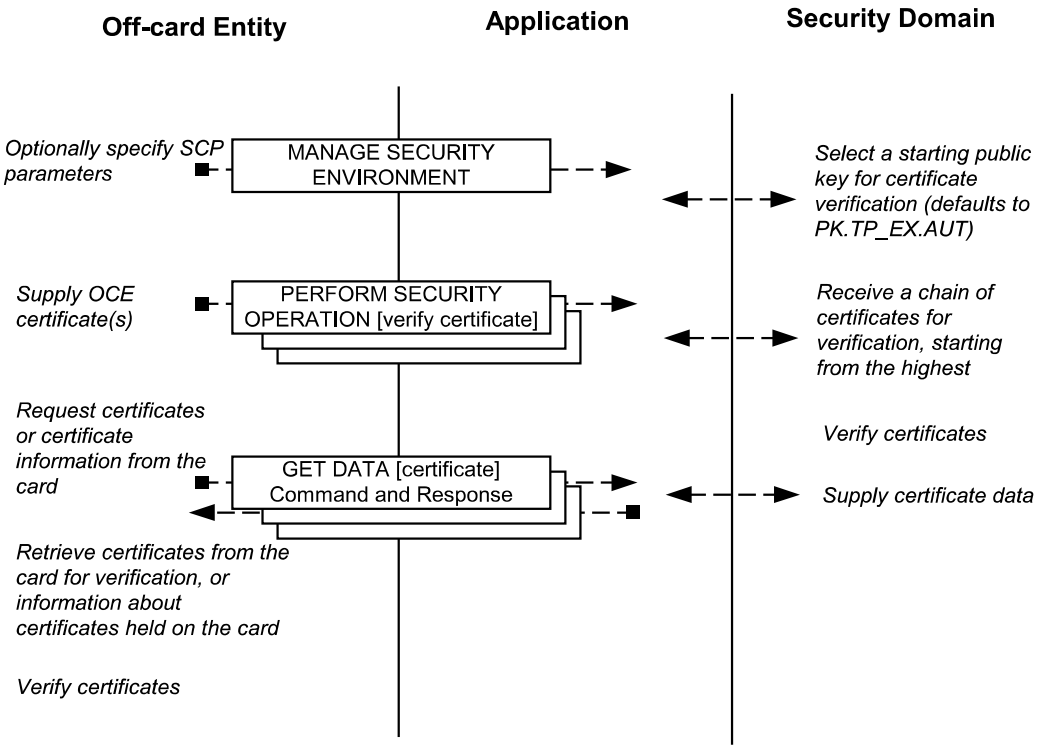


Figure F-4 - Certificate Verification Flow

The following commands shall be supported:

- MANAGE SECURITY ENVIRONMENT containing a 'cryptographic mechanism reference' designating the GlobalPlatform Secure Channel Protocol '10', see section F.4.5 for further details;
- PERFORM SECURITY OPERATION [verify certificate] command, see section F.4.7 for further details;
- GET DATA [certificate] command, see section F.4.3 for further details.

On receipt of a MANAGE SECURITY ENVIRONMENT command, any existing Secure Channel Session (on the same logical channel of the same card I/O interface) shall be terminated, regardless of the validity

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0259

of the command: both the Current Security Level and Session Security Level are reset to NO_SECURITY_LEVEL. The MANAGE SECURITY ENVIRONMENT command may refer to a previously validated public key. If the MANAGE SECURITY ENVIRONMENT command is omitted, Security Domain default values and options shall apply, in particular for the initial default public key to use in subsequent command processing.

On receipt of a PERFORM SECURITY OPERATION [verify certificate] command not preceded by a MANAGE SECURITY ENVIRONMENT or another PERFORM SECURITY OPERATION [verify certificate] command, any existing Secure Channel Session (on the same logical channel of the same card I/O interface) shall be terminated, regardless of the result of the certificate verification: both the Current Security Level and Session Security Level are reset to NO_SECURITY_LEVEL.

Multiple PERFORM SECURITY OPERATION [verify certificate] commands may be received. In this case, a chain of certificates is presented to the Security Domain and the certificate contained in each command is verified using the Current Public Key. The Current Public Key is the public key that the Security Domain validated when verifying the last certificate presented by the Off-Card Entity during the same Secure Channel Session initiation. The Security Domain shall have a default Current Public Key at the start of a Secure Channel Session initiation phase: Trust Point for External Authentication (PK.TP_EX.AUT).

Any failure in verifying an Off-Card Entity's certificate aborts the current Secure Channel Session initiation phase, and any public keys validated during that initiation phase shall be discarded.

Multiple GET DATA [certificate] commands may be received. They may be interleaved with PERFORM SECURITY OPERATION [verify certificate] commands. The Security Domain makes no assumptions about whether the Off-Card Entity has obtained enough certificates in order to validate the Security Domain's public key. The Off-Card Entity may use the GET DATA [certificate] command for EF.OD to retrieve information about the different certificates the Security Domain holds, see F.1.2.3 for further details on EF.OD.

F.1.3.3 Certificate Information

The Security Domain shall support access to EF.OD with a 'data object id' set to '5031'. EF.OD identifies each certificate that can be retrieved by the Off-Card Entity for verification. EF.OD contains a set of Cryptographic Information Objects (CIOs) as defined in ISO/IEC 7816-15, each of which describes a certificate and gives a pointer (in the form of a 'data object id') to the certificate data present within the Security Domain. The contents of EF.OD and its consistency with the certificates actually present within the Security Domain are the responsibility of the Security Domain's Provider and beyond the control of the card.

According to ISO/IEC 7816-15, an individual CIO for a certificate may be coded as follows (xxCertificate being of type CertificateChoice):

```
xxCertificate:
{
 commonObjectAttributes
      {
      label                Label              OPTIONAL
      flags                CommonObjectFlags OPTIONAL,
      authId               Identifier  OPTIONAL,
      userConsent          INTEGER (1..cia-ub-userConsent)       OPTIONAL,
      accessControlRules   SEQUENCE SIZE (1..MAX) OF AccessControlRule OPTIONAL,
      ….},
classAttributes
      {
      id              Identifier,
      authority       BOOLEAN     DEFAULT FALSE,
```

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0260

```
        identifier      CredentialIdentifier {{KeyIdentifiers}}      OPTIONAL,
        certHash        [0] CertHash    OPTIONAL,
        trustedUsage    [1] Usage OPTIONAL
        identifiers     [2] SEQUENCE OF CredentialIdentifier {{KeyIdentifiers}} OPTIONAL,
        validity        [4] Validity    OPTIONAL,
        ….},
    typeAttributes
        {
        value           ObjectValue {Certificate},
        ….}
}
```

where typeAttributes vary per type of CertificateChoice, for instance x509CertificateAttributes are:

```
    typeAttributes
        {
        value           ObjectValue {Certificate},
        subject         Name            OPTIONAL,
        issuer          [0] Name  OPTIONAL,
        serialNumber    CertificateSerialNumber    OPTIONAL,
        ….}
```

F.1.3.4 Certificate Formats

Certificate contents and encoding are outside the scope of this specification. Examples are given for illustration only. Certificates may contain various data objects and elements as defined in ISO/IEC 7816-4, 7816-6 and 7816-8 (using application tagging), ITU Recommendation X.509 (universal and context-specific tagging) and elsewhere. The data objects included are defined by the certificate issuer (CA).

The following table identifies some data items that appear in certificates - 'presence' indicates when the item can be expected to be present:

Name as used in X.509	Name as used in 7816-8	Application Tag assigned by 7816	Presence
Issuer	Issuer Identification Number	'42' (7816-8)	Always
Subject	Certificate holder reference OR Cardholder name	'5F20' (7816-8)	Always
SubjectPublicKey	Cardholder public key	'5F49' or '7F49' (7816-8)	Always
KeyUsage	Certificate holder authorization	'5F4C' (7816-9)	As required
CertificateSerialNumber	Certificate serial number	-	Always
n/a	Certificate contents	'5F4E' (7816-8)	Non self descriptive certificates

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0261

Name as used in X.509	Name as used in 7816-8	Application Tag assigned by 7816	Presence
Signature	Static internal authorization OR Digital signature	'9E' (7816-8)	Always
n/a	Public key remainder	'5F38' (7816-6)	Signature scheme with recovery

Table F-2: Example of Data Included in Certificates

Certificates may either be self descriptive, where the signature is across individual data objects; or non self descriptive, where the signature is across concatenated value fields, without tags and lengths. Whether the certificate to be verified by the Security Domain is self-descriptive or non-self descriptive must be indicated in the PERFORM SECURITY OPERATION [verify certificate] command.

A Security Domain is only required to handle a single certificate format throughout a chain of certificates to be verified by the Security Domain.

Tag '67' in Card Recognition Data or Security Domain Management Data may contain one or more OIDs identifying the Security Domain's Trust Point's certification policy and/or the format of certificates that can be verified by the Security Domain and/or the format of certificates that can be retrieved from the Security Domain. They may also identify the cryptographic algorithms used for certificates, unless they are indicated in the certificates themselves.

F.1.3.4.1 Certificate without Message Recovery

The data and public key to be certified are concatenated and a digest is created. A signature block is then prepared containing the digest and any necessary padding, constant and random data. The format of the signature block is defined by the certificate issuer (CA). The signature block size depends on the asymmetric algorithm and the certifying key size. The signature block is then signed using the certificate issuer's private certifying key, to form the value field of the Certificate Signature data object. The result is a certificate typically containing the data and public key to be certified and the Certificate Signature.

Certificate without Message Recovery, Self Descriptive Certificate

The data objects to be certified, including the public key data object, are TLV encoded and are concatenated TLV-encoded before the digest is created and the signature block prepared. The result of the signing operation is the Certificate Signature. The certificate is TLV encoded as a series of data objects, containing all the data objects to be certified, including the public key data object and the Certificate Signature data object. X.509 certificates fall into this category.

The formation of the certificate is shown in the following example:

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0262

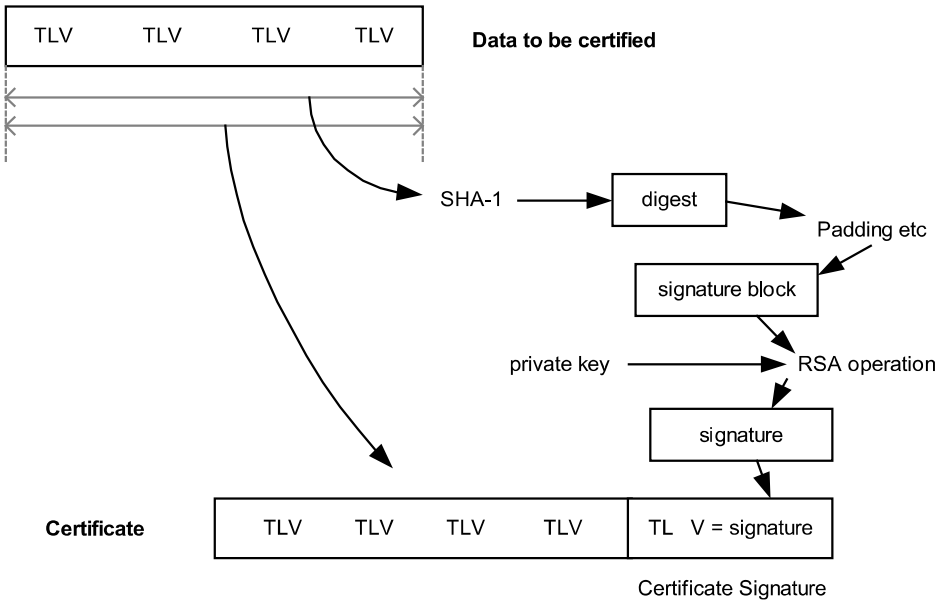


Figure F-5: Certificate Formation - Self Descriptive Certificate Without Message Recovery

Certificate without Message Recovery, Non-Self Descriptive Certificate

The data and public key to be certified are a concatenated set of value fields. The format of the value fields included is defined by the certificate issuer (CA) and implicitly known to the recipients. A digest of the concatenated value fields is created and the signature block prepared. The result of the signing operation is the Certificate Signature. The certificate contains two data objects: the Certificate Contents, whose value is the concatenated set of value fields being certified, and the Certificate Signature.

The formation of the certificate is shown in the following example:

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0263

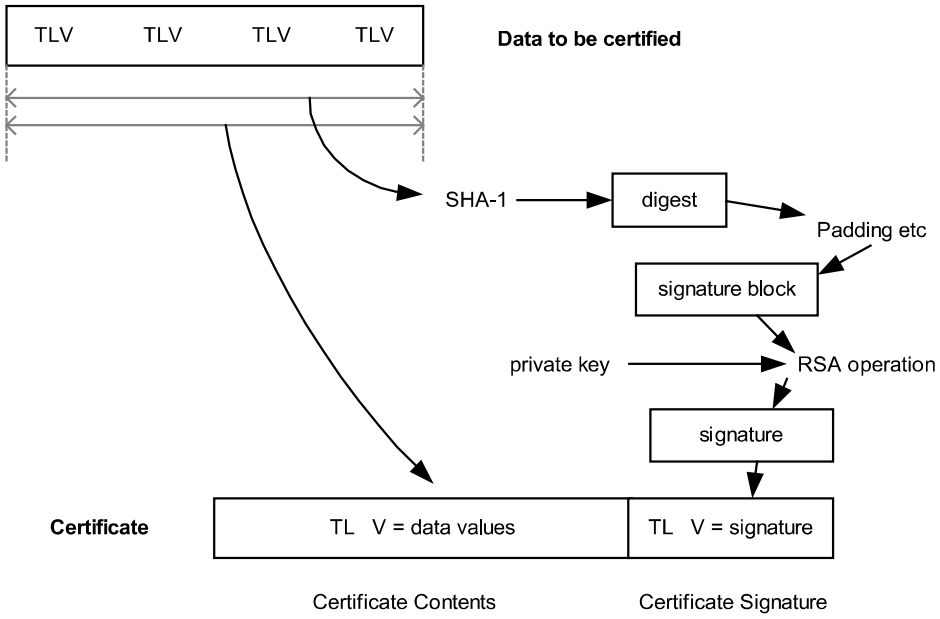


Figure F-6: Certificate Formation - Non Self Descriptive Certificate Without Message Recovery

F.1.3.4.2 Certificate with Message Recovery

The data and public key to be certified is concatenated and a digest is created. A signature block is then prepared containing the digest, as much of the concatenated set of data and public key to be certified as can be included in the block and any necessary padding, constant and random data. The format of the signature block is defined by the certificate issuer (CA). The signature block size depends on the asymmetric algorithm and the certifying key size. The signature block is then signed using the certificate issuer's private certifying key, to form the value field of the Certificate Signature data object. Any part of the public key that could not be included in the Certificate Signature is then included in the value field of a separate Public Key Remainder data object. The result is a certificate typically containing two data objects: the Public Key Remainder and the Certificate Signature.

Certificate with Message Recovery, Self Descriptive Certificate

The data objects to be certified, including the public key data object, are TLV encoded and are concatenated TLV-encoded before the digest is created and the signature block prepared. The result of the signing operation is the Certificate Signature.

The formation of the certificate is shown in the following diagram:

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0264

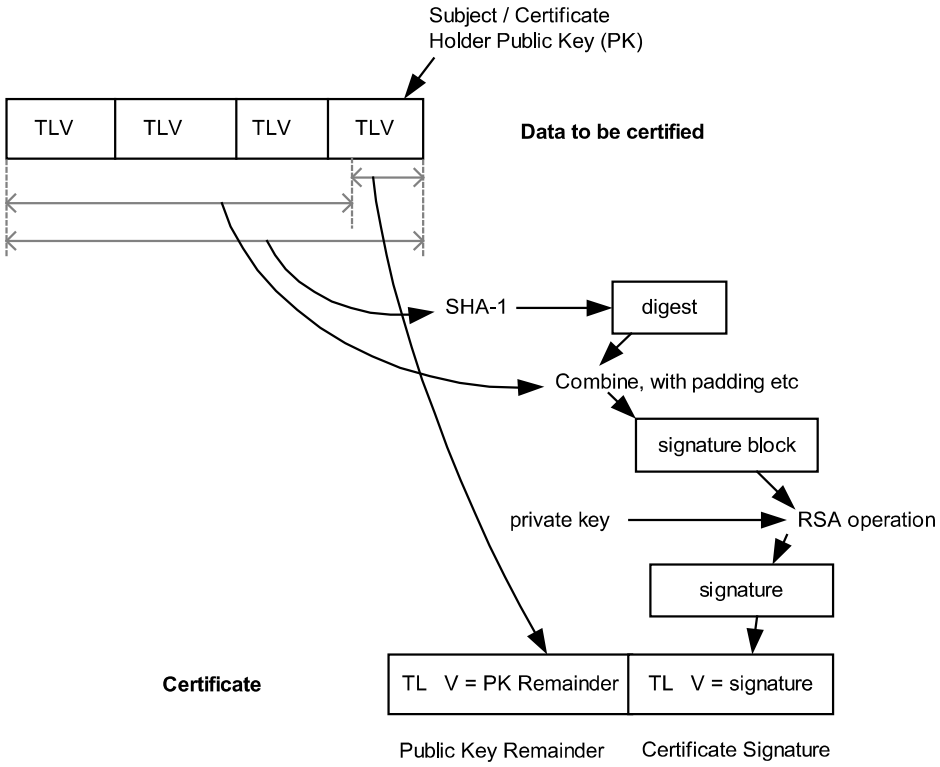


Figure F-7: Certificate Formation - Self Descriptive Certificate with Message Recovery

Certificate with Message Recovery, Non-Self Descriptive Certificate

The data and public key to be certified is a concatenated set of value fields. The format of the value fields included is defined by the certificate issuer (CA) and implicitly known to the recipients. A digest of the concatenated value fields is created and the signature block prepared. The result of the signing operation is the Certificate Signature.

The formation of the certificate is shown in the following diagram:

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0265

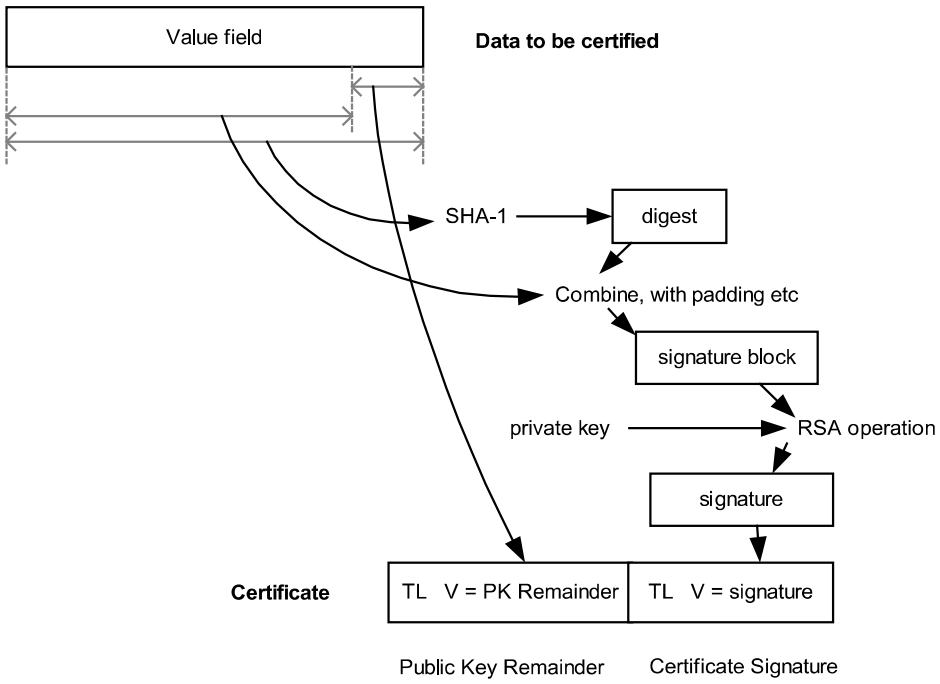


Figure F-8: Certificate Formation - Non Self Descriptive Certificate with Message Recovery

F.1.4 Entity Authentication

F.1.4.1 Overview of Entity Authentication

Once the parties have validated each other's public key, the Security Domain shall authenticate the Off-Card Entity using a challenge / response mechanism, and the Off-Card Entity may also authenticate the Security Domain in the same way.

Entity authentication of the Security Domain is optional, at the discretion of the Off-Card Entity. However, if the INTERNAL AUTHENTICATE command is required for use in session key agreement, then authentication of the Security Domain is performed.

F.1.4.2 Entity Authentication Process Flow

The following diagram gives an overview of the flow for Entity Authentication.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0266

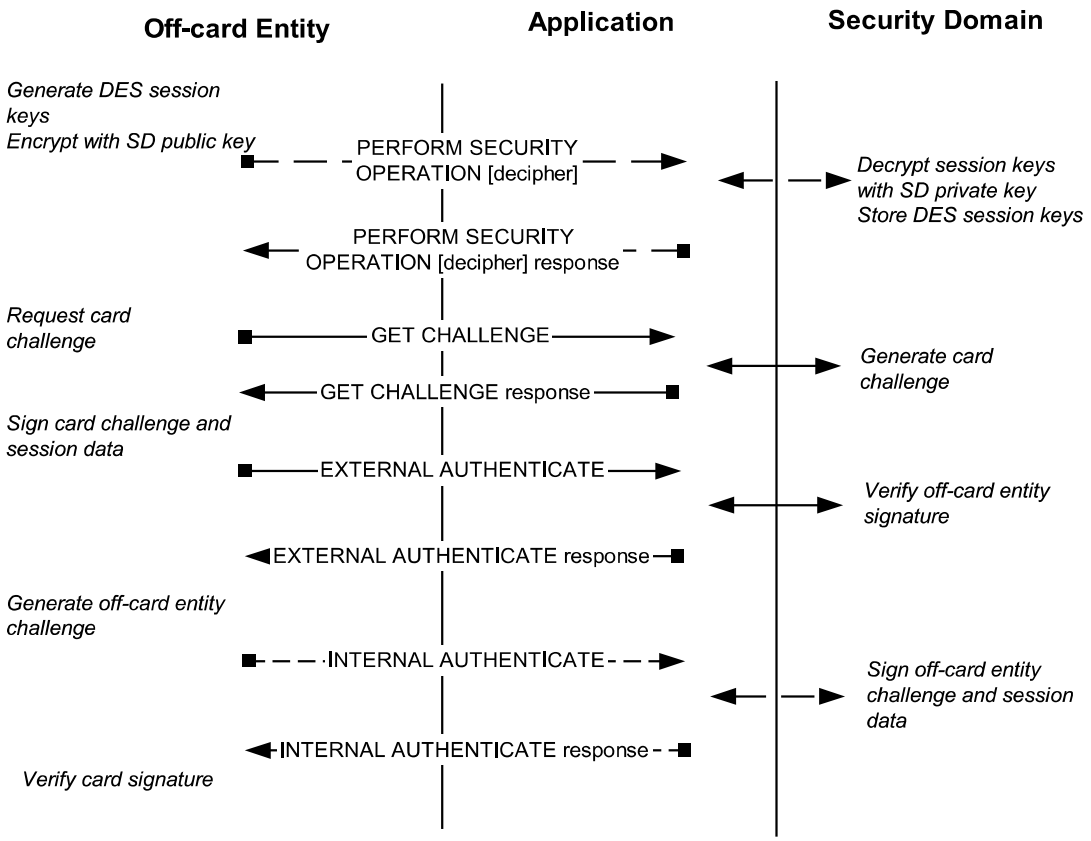


Figure F-9: Entity Authentication Flow

The following commands shall be supported:

- PERFORM SECURITY OPERATION [decipher] command, see section F.4.6 for further details;
- GET CHALLENGE command, see section F.4.2 for further details;
- EXTERNAL AUTHENTICATE command, see section F.4.1 for further details;
- INTERNAL AUTHENTICATE command, see section F.4.4 for further details.

The PERFORM SECURITY OPERATION [decipher] command is optional but shall be executed when value '02' of parameter "i" is supported - see section F.1.1.

The successful processing of the EXTERNAL AUTHENTICATE command (i.e. successful verification of the Off-Card Entity's signature) shall set both the Current Security Level and Session Security Level according to the rules described in section F.1.4.2 - *Security Level Establishment*. The failed processing of the EXTERNAL AUTHENTICATE command (i.e. unsuccessful verification of the Off-Card Entity's signature) shall reset both the Current Security Level and Session Security Level to NO_SECURITY_LEVEL and all the public keys previously verified within the current initiation phase shall be discarded.

The INTERNAL AUTHENTICATE command is optional for the key transport option and mandatory for key agreement. The INTERNAL AUTHENTICATE command shall only be received and processed once

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0267

during Secure Channel Session initiation. Any error in the sequence flow or signing operation must abort the current Secure Channel Session: both the Current Security Level and Session Security Level shall be reset to NO_SECURITY_LEVEL.

F.1.4.3 Security Domain Authentication

The format and contents of Security Domain signature vary depending on the options chosen, as follows:

Key transport, signature without message recovery

The Security Domain signature is the result of generating a digest (hash) over a set of data, creating a signature block, and signing the signature block with the Security Domain private key (SK.SD.AUT). The data to be hashed and the contents of the signature block are as shown below.

Name	Length	Value	Presence
Session Key(s)	n x (16 or 24)	'xxxx...'	Mandatory
Off-Card Entity challenge	16	'xxxx...'	Mandatory

Table F-3: Data to Hash

Session Keys shall be in the same order as provided in the PERFORM SECURITY OPERATION [decipher] command: see sections F.3.1.2 and F.4.7.

Name	Length	Value	Presence
Padding	2	'0001'	Mandatory
Padding (‘FF’)	8-n	'FF'...'FF'	Mandatory
Padding (‘00’)	1	'00'	Mandatory
DER encoded digest algorithm id - encoded as an object identifier	Variable	'xxxx...' (see section F.2.2)	Mandatory
DER encoded Hash (length and contents depend on the digest algorithm)	Variable	'xxxx...'	Mandatory

Table F-4: Security Domain Signature Block

Key agreement, signature with message recovery

The Security Domain signature is the result of generating a digest (hash) over a set of data, creating a signature block, and signing the signature block with the Security Domain private key (SK.SD.AUT). The data to be hashed and the contents of the signature block are as shown below.

Name	Length	Value	Presence
Random Padding (RP)	1-n	Same value as RP in Table F-6	Mandatory
Card Secret (CS)	32	Same value as CS in Table F-6	Mandatory
Off-Card Entity challenge	8	'xxxx...'	Mandatory
Off-Card Entity id	8	'xxxx...' (part of CERT.OCE.AUT)	Mandatory

Table F-5: Data to Hash

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0268

Name	Length	Value	Presence
Padding	1	'6A'	Mandatory
Random Padding (RP)	1-n	Same value as RP in Table F-5	Mandatory
Card Secret (CS)	32	Same value as CS in Table F-5	Mandatory
Hash	20	'xxxx...'	Mandatory
Padding	1	'BC'	Mandatory

Table F-6: Security Domain Signature Block

The Security Domain signature is encrypted to ensure that the Card Secret is not divulged. To do this, the minimum of the values SIG.SD.AUT and (N.PK.SD.AUT – SIG.SD.AUT) is encrypted with the Off-Card Entity public key (PK.OCE.AUT), where N.PK.SD.AUT denotes the modulus of the Security Domain public key. This ensures that the data to be encrypted is always smaller than the modulus of the Off-Card Entity public key. Note that the modulus of the Security Domain public key and modulus of the Off-Card Entity public key must have the same length in bits. See ISO/IEC 9796-2, Digital Signature scheme 1.

F.1.4.4 Off-Card Entity Authentication

The format and contents of Off-Card Entity signature vary depending on the option chosen, as follows:

Key transport, signature without message recovery

The Off-Card Entity signature is the result of generating a digest (hash) over a set of data, creating a signature block, and signing the signature block with the Off-Card Entity private key (SK.OCE.AUT). The data to be hashed and the contents of the signature block are as shown below.

Name	Length	Value	Presence
Tag of Security Level	1	'D3'	Mandatory
Length of Security Level	1	'01'	Mandatory
Security Level	1	'xx'	Mandatory
Control reference template (CRT) tag	1	'B4' or 'B8'	Mandatory
Length of CRT	1	'00' - '7F'	Mandatory
CRT for Session Key(s)	n	'xxxx...'	Mandatory
…	…	…	…
CRT tag	1	'B4' or 'B8'	Optional
Length of CRT	1	'00' - '7F'	Conditional
CRT for Session Key(s)	n	'xxxx...'	Conditional
Card challenge	16		Mandatory

Table F-7: Data to Hash

The control reference templates (CRTs) include session keys, and must be in the same order as provided in the PERFORM SECURITY OPERATION [decipher] command - see Table F-32.

Name	Length	Value	Presence
Padding ('0001')	2	'0001'	Mandatory
Padding ('FF')	8-n	'FF'	Mandatory

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0269

Name	Length	Value	Presence
Padding ('00')	1	'00'	Mandatory
DER encoded digest algorithm id (encoded as an object identifier)	Variable	'xxxx...' (see section F.2.2)	Mandatory
DER encoded Hash (length and contents depend on the digest algorithm)	Variable	'xxxx...'	Mandatory

Table F-8: Off-Card Entity Signature Block

Control reference templates (CRTs) include session keys, and must be in the same order as provided in the PERFORM SECURITY OPERATION [decipher] command.

The format and contents of Card Signature vary depending on the options chosen, as discussed below.

Key agreement, signature with message recovery

The Off-Card Entity signature is the result of generating a digest (hash) over a set of data, creating a signature block, and signing the signature block with the Off-Card Entity private key (SK.OCE.AUT). The data to be hashed and the contents of the signature block are as shown below.

Name	Length	Value	Presence
Random Padding (RPD)	1-n	Same value as RPD in Table F-10	Mandatory
Tag of Security Level	1	'D3'	Mandatory
Length of Security Level	1	'01'	Mandatory
Security Level	1	'xx'	Mandatory
CRT tag	1	'B4' or 'B8'	Mandatory
Length of CRT	1	'00' - '7F'	Mandatory
CRT for session key(s)	n	'xxxx...'	Mandatory
…	…	…	…
CRT tag	1	'B4' or 'B8'	Optional
Length of CRT	1	'00' - '7F'	Conditional
CRT for Session Key(s)	n	'xxxx...'	Conditional
Off-Card Entity Secret (OES)	32	Same value as OES in Table F-10	Mandatory
Card challenge	8	'xxxx...'	Mandatory
Card id: TBD (part of CERT.SD.AUT)	8	'xxxx...'	Mandatory

Table F-9: Data to Hash

Control reference templates (CRTs) exclude session keys, and must be in the same order as provided in signature block.

Name	Length	Value	Presence
Padding	1	'6A'	Mandatory
Random Padding (RPD)	1-n	Same value as RPD in Table F-9	Mandatory
Tag of Security Level	1	'D3'	Mandatory

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0270

Name	Length	Value	Presence
Length of Security Level	1	'01'	Mandatory
Security Level	1	'xx'	Mandatory
CRT tag	1	'B4' or 'B8'	Mandatory
Length of CRT	1	'00' - '7F'	Mandatory
CRT for session key(s)	1-n	'xxxx...'	Mandatory
…	…	…	…
CRT tag	1	'B4' or 'B8'	Optional
Length of CRT	1	'00' - '7F'	Conditional
CRT for Session Key(s)	n	'xxxx...'	Conditional
Off-Card Entity Secret (OES)	32	Same value as OES in Table F-9	Mandatory
Hash	20	'xxxx...'	Mandatory
Padding	1	'BC'	Mandatory

Table F-10: Off-Card Entity Signature Block

Control reference templates (CRTs) exclude session keys, and must be in the same order as input to the hash function.

The Off-Card Entity signature is encrypted to ensure that the Off-Card Entity secret is not divulged. To do this, the minimum of the values SIG.OCE.AUT and (N.PK.OCE.AUT – SIG.OCE.AUT) is encrypted using the Security Domain public key (PK.SD.AUT). N.PK.OCE.AUT denotes the modulus of the Off-Card Entity public key. This ensures that the data to be encrypted is always smaller than the modulus of the Security Domain public key. Note that the modulus of the Security Domain public key and modulus of the Off-Card Entity public key must have the same length in bits. See ISO/IEC 9796-2, Digital Signature scheme 1.

F.1.5 Session Key and Security Level Establishment

When using the key transport option, Entity Authentication is preceded by the session keys and requested Security Level being sent to the Security Domain using the PERFORM SECURITY OPERATION [decipher] command; the Security Domain stores them until session initiation is complete.

A Security Domain supporting the key transport option shall decrypt with its private key (SK.SD.AUT or if the Security Domain has more than one key pair, the private key identified in the MANAGE SECURITY ENVIRONMENT command) the command data field of the PERFORM SECURITY OPERATION [decipher] command. Any failure in the decryption operation aborts the current Secure Channel Session initiation phase, and any public keys validated during that initiation phase shall be discarded.

In the key transport option the Secure Channel Session is established after successful processing of the EXTERNAL AUTHENTICATE command. An INTERNAL AUTHENTICATE command can be issued immediately after the EXTERNAL AUTHENTICATE command without secure messaging.

In the key agreement option the Secure Channel Session is established after successful processing of the EXTERNAL AUTHENTICATE and INTERNAL AUTHENTICATE commands.

F.1.5.1 Session Key Establishment

The Off-Card Entity supplies the Security Domain with details of what session keys are to be established. This information is in the form of 'control reference templates' (see section F.3.1.2) which are supplied

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0271

either in the PERFORM SECURITY OPERATION [decipher] command (with the key transport option) or in the EXTERNAL AUTHENTICATE command (with the key agreement option).

If the key transport option is used, then the session keys are provided by the Off-Card Entity within the 'control reference templates' (see section F.3.1.2).

If the key agreement option is used, then the secrets exchanged between the Off-Card Entity and the Security Domain during the Entity Authentication process are used to establish session keys, as defined in section F.3.1 - *DES Session Keys*.

Once session keys have been established successfully, ICV sequence counter(s), used for secure messaging on subsequent commands and responses, are initialized as described in section F.3.2 - *Secure Messaging*.

F.1.5.2 Security Level Establishment

The requested Security Level is supplied in the PERFORM SECURITY OPERATION [decipher] command (with the key transport option) or in the EXTERNAL AUTHENTICATE command (with the key agreement option).

The successful initiation of a Secure Channel Session shall set the Current Security Level and Session Security Level to the requested Security Level combined with the AUTHENTICATED or ANY_AUTHENTICATED indicator (see section 10.4.2 - *Authentication with asymmetric cryptography* for further details). If the requested Security Level is set to zero, the successful initiation of a Secure Channel Session shall set the Current Security Level and Session Security Level to AUTHENTICATED or ANY_AUTHENTICATED only.

F.1.6 Protocol Rules

The Current Security Level of a communication not included in a Secure Channel Session shall be set to NO_SECURITY_LEVEL. In accordance with the general rules described in chapter 10 - *Secure Communication*, the following rules shall apply:

- The successful initiation of a Secure Channel Session shall set the Current Security Level to the requested Security Level from the selected Application's perspective: it is at least set to AUTHENTICATED or ANY_AUTHENTICATED (see section 10.4.2 - *Authentication with asymmetric cryptography* for details;
- 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, then:
  - 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 for applying its own security rules.
- If a Secure Channel Session is active for incoming commands (i.e. Current Security Level at least set to either AUTHENTICATED or ANY_AUTHENTICATED), the security of the incoming command shall be checked according to the Current Security Level, or if the APDU class byte indicates Secure Messaging and Secure Messaging data objects are present in the command data field:
  - When the security of the command does not match or exceed 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;

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0272

- 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;
- If (one of) the appropriate session key(s) is not available, 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 shall reflect the level of security established by the current command (e.g. C-MAC and/or C-ENCRYPTION). The Application is responsible for further processing the command.

- If a Secure Channel Session is active for outgoing responses (i.e. Current Security Level at least set to AUTHENTICATED or ANY_AUTHENTICATED), secure messaging protection shall be applied to the outgoing response according to the Current Security Level (i.e. R-MAC and/or R-ENCRYPTION):
  - If a cryptographic error occurs, a security error shall be returned, the Secure Channel Session aborted and the Current Security Level reset to NO_SECURITY_LEVEL;
  - If (one of) the appropriate session key(s) is not available, a security error shall be returned, the Secure Channel Session aborted and the Current Security Level reset to NO_SECURITY_LEVEL;
  - Otherwise, the Secure Channel Session shall remain active and the Current Security Level unmodified.
- If a Secure Channel Session is aborted, it is still considered not terminated;
- If the Security Domain supports application data encryption and/or decryption, it shall decrypt or encrypt a block of secret data upon request. If the service is not supported or if (one of) the appropriate cryptographic key(s) is not available, the request shall be rejected but the Current Security Level, Session Security Level and Secure Channel Session in operation (if any) shall not be impacted;
- The current Secure Channel Session shall be terminated (if aborted or still open), both the Current Security Level and Session Security Level reset to NO_SECURITY_LEVEL on either:
  - Attempt to initiate a new Secure Channel Session;
  - 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).

F.2 Cryptographic Algorithms

The cryptographic and hashing algorithms described in appendix B - *Algorithms (Cryptographic and Hashing)* apply to SCP10. This section defines the additional requirements for SCP10.

F.2.1 Asymmetric cryptography

In this appendix, signing means the deciphering of a signature block using the signer's private RSA key.

For certificates to be verified by the card, and message signatures signed and verified by the card, the cryptographic scheme shall be RSA. The signature block and signature are the same length as the key modulus.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0273

In Entity Authentication, Digital Signature Scheme 1 in ISO/IEC 9796-2 shall be used for signature with message recovery, and the signature scheme with appendix RSASSA-PKCS1-V1_5 in PKCS#1 v2.1 for signature without message recovery.

The details of the signature scheme for certificates shall be either implicitly known by the Off-Card Entity or specified by the Security Domain Trust Point through the contents of Card Recognition Data or Security Domain Management Data in tag '67'.

In the key transport option, the cryptographic scheme for encrypting the session keys and their CRT templates shall be RSA according to the encryption scheme RSAES-PKCS1-v1_5 as defined in PKCS#1 v2.1.

F.2.2 Digest Algorithm

The default digest algorithm for use in conjunction with SCP10 asymmetric cryptography in Entity Authentication shall be SHA-1 for this version of the Specification. An alternative algorithm may be specified in Card Recognition Data or Security Domain Management Data in tag '67'.

The Object Identifier for SHA-1 is:

{iso(1) identified-organization(3) oiw(14) secsig(3) algorithms(2) 26}

which is DER-TLV encoded as '2B 0E 03 02 1A'.

F.2.3 Message Integrity ICV

The ICV for the each C-MAC and R-MAC calculation is obtained by enciphering an ICV sequence counter. The ICV sequence counter is initialized during Secure Channel initiation to either:

- the value supplied in tag '91' of the control reference template, for the key transport option (separate initial value for each key), or
- the concatenation of the last 4 bytes of the card secret and the last 4 bytes of Off-Card Entity secret, for the key agreement option (same initial value for all keys).

The ICV sequence counter is incremented by one for each C-MAC and R-MAC. For calculating a C-MAC ICV, the ICV sequence counter is single-DES enciphered using the first part of the Secure Channel C-MAC key. For calculating an R-MAC ICV, the ICV sequence counter is single-DES enciphered using the first part of the Secure Channel R-MAC key.

F.2.4 Message Integrity C-MAC and R-MAC

Message integrity is achieved by applying a MAC to message data. The MAC may be:

- C-MAC for APDU command messages (generated by the Off-Card Entity);
- R-MAC for APDU response messages (generated by the card).

The receiving entity, on receipt of the message containing a MAC, using the same session key, performs the same operation and by comparing its generated MAC with the MAC received from the sending entity is assured of the integrity of the full command or response.

The integrity of the sequence of APDU command or response messages being transmitted to the receiving entity is achieved by using an encrypted sequence counter as part of the MAC generation. This ensures the receiving entity that all messages in a sequence have been received.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0274

F.2.5 APDU Encryption and Decryption for Message Confidentiality

Message confidentiality is achieved by encrypting the whole of the command or response data field. This includes any data within the data field that has already been protected for another purpose, such as secret or private keys encrypted with the data encryption key.

F.3 Cryptographic Usage

F.3.1 DES Session Keys

F.3.1.1 Overview

All session keys shall be double or triple length DES keys.

The Off-Card Entity supplies information on cryptographic keys to be established for the session in a set of control reference templates. Each control reference template specifies the usage of the key.

In the case of key transport, the templates are supplied in the PERFORM SECURITY OPERATION [decipher] command and contain the actual key values.

In the case of key agreement, the templates are supplied in the EXTERNAL AUTHENTICATE command, and do not contain the key values.

F.3.1.2 Control Reference Templates

A control reference template (CRT) is structured as follows:

Tag	Length	Name	Presence
‘B4’ or ‘B8’	'00' - '7F'	CRT tag = ‘B4’ (CCT) or ‘B8’ (CT)	Mandatory
'95'	1	Key Usage Qualifier = ‘10’ (secure messaging for commands), ‘20’ (secure messaging for responses), ‘30’ (secure messaging for commands and responses), ‘40’ (encipherment of sensitive data in responses), ‘80’ (encipherment of sensitive data in commands), ‘C0’ (encipherment of sensitive data in commands & responses)	Mandatory
'80'	0 or 1	Optional cryptographic mechanism. Contains Key Type, coded according to Table 11-16	Optional
'D1'	0, 16 or 24	Off-Card Entity Session Key	Conditional
'91'	0 or 8	Initial value of sequence counter, for use in secure messaging	Conditional

Table F-11: Single CRT

Note: Key Usage Qualifier and the CRT tag together define the key usage, which is C-MAC and/or R-MAC for control reference template 'B4', and C-ENC and/or R-ENC or DEK for control reference template 'B8'.

The Key Type identifies the cryptographic algorithm.

The Off-Card Entity Session Key and sequence counter data objects are only present with the key transport option.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0275

The sequence counter data object is used in secure messaging as the initial value for the ICV sequence counter for this key, which is encrypted as defined in section F.2.3 - *Message Integrity ICV* to derive the ICV for the first MAC generated using this key.

F.3.1.3 Session Key Derivation

With the key agreement option, session keys are derived from the secret data exchanged during Secure Channel initiation as follows:

- The two 32-byte secrets Off-Card Entity Secret and Card Secret are exclusive or-ed, giving result (1);
- A 32 byte binary counter is set to a value depending on the key usage and its position in the set of CRTs supplied, as shown below;
- Result (1) is appended with a 32 bit binary counter with the appropriate value for this key, and the result is hashed using SHA-1, giving result (2);
- Bytes 1-16 of result (2) form the double-length DES session key.

Counter value	Key
1	The MAC key whose CRT is first in the set of CRTs supplied by the Off-Card Entity
2	The ENC key whose CRT is first in the set of CRTs supplied by the Off-Card Entity
3	A subsequent MAC key, if any
4	A subsequent ENC key, if any
5	The data encryption key whose CRT is first in the set of CRTs supplied by the Off-Card Entity
6	A subsequent data encryption key, if any

Table F-12: Counter Value for Session Key Calculation

F.3.2 Secure Messaging

F.3.2.1 APDU Command C-MAC Protection

This section applies where command integrity (C-MAC) is required but not command confidentiality (C-ENC).

A C-MAC is generated by an Off-Card Entity and applied across the full APDU command being transmitted to the card including the header, the command data field (if present) and Le (if present). Input data to the MAC calculation is first prepared as defined in ISO/IEC 7816-4:

- The following data is concatenated:
  - The command header CLA, INS, P1, P2 from the unprotected APDU, with the logical channel bits in the CLA byte set to zero, and appended with four bytes '80 00 00 00';
  - If a command data field is present in the unprotected APDU, a BER-TLV data object with tag '81' containing the complete original command data field , regardless of its contents and format;

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0276

- If Le is present in the unprotected APDU, a BER-TLV data object with tag '97' containing the original Le value;

- DES padding is applied as defined in appendix B.4 - *DES Padding*.

A C-MAC is generated using the Secure Channel C-MAC session key, the encrypted sequence counter as the ICV as defined in section F.2.3, and the signature method described in appendix B.1.2.2 - *Single DES Plus Final Triple DES MAC* across the input data.

To reflect the presence of a C-MAC in the command message, the unprotected APDU shall be modified as follows:

- The class byte shall be modified to indicate that this APDU command includes secure messaging. This is achieved by setting to '11' bits 4-3 of a class byte indicating a logical channel number 0 to 4 (unprotected CLA set to '00' - '03' or '80' - '83') or by setting to '1' bit b6 of a class byte indicating a logical channel number 4 to 19 (unprotected CLA set to '40' - '4F' or 'C0' - 'CF'): see section 11.1.4. The logical channel bits are unchanged;
- The length of the command message (Lc) shall be incremented by:
  - 10 bytes to allow for the C-MAC data object, plus
  - 2 or more bytes to allow for the tag and length of the command data field data object (if command data is present - note: the length field may be longer than one byte), plus
  - 3 bytes to allow for the Le data object (if Le is present).
- The command data, if present in the unprotected APDU, shall be encapsulated in a BER-TLV data object with tag '81' and a length field coded according to ISO 8825-1;
- The Le byte, if present in the unprotected APDU, shall be contained in a BER-TLV data object with tag '97';
- The C-MAC shall be encapsulated in a BER-TLV data object with tag '8E' and appended at the end of the command data field.

No padding is present in the transmitted APDU.

The following diagram shows the message reformatting that is performed by the Off-Card Entity when a command is protected for integrity.

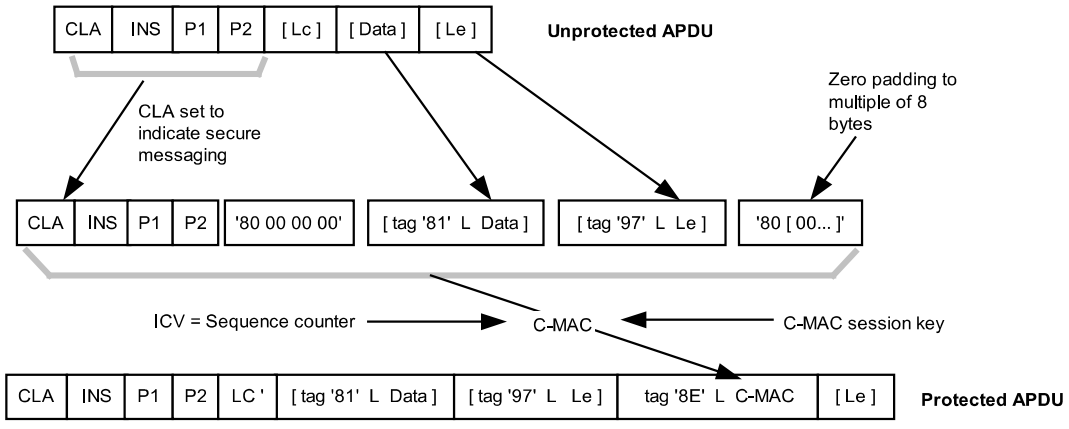


Figure F-10: APDU C-MAC Generation

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0277

The card, in order to verify the C-MAC, shall perform the same procedure as employed by the Off-Card Entity in order to verify the C-MAC. The ICV sequence counter used in ICV calculation is then incremented. This is true regardless of whether the APDU processing completes successfully or not i.e. a new sequence counter value shall always be used for the next C-MAC or R-MAC.

F.3.2.2 APDU Command C-ENC Protection

This section applies where command confidentiality (C-ENC) is required but not command integrity (C-MAC).

No encryption shall be applied to a command where there is no command data field: in this case the command message (header and optional Le) is sent without modification.

Otherwise the Off-Card Entity encrypts the command data field of the command message being transmitted to the card. This includes any data within the data field that has already been protected for another purpose e.g. secret or private keys encrypted with the data encryption session key.

Prior to encrypting the data, DES padding is applied as defined in appendix B.4 - *DES Padding*.

The padded command data field is enciphered using triple DES in CBC mode as defined in appendix B.1.1.1, the C-ENC session key established during the Secure Channel initiation process and an ICV of zero.

To reflect the C-ENC protection of the command, the unprotected APDU shall be modified as follows:

- The class byte shall be modified to indicate that this APDU command includes secure messaging. This is achieved by setting to '10' bits 4-3 of a class byte indicating a logical channel number 0 to 4 (unprotected CLA set to '00' - '03' or '80' - '83') or by setting to '1' bit b6 of a class byte indicating a logical channel number 4 to 19 (unprotected CLA set to '40' - '4F' or 'C0' - 'CF'): see section 11.1.4. The logical channel bits are unchanged;
- Lc shall be incremented by:
  - 4 or more bytes to allow for the tag and length of the command data field data object, the padding indicator and the variable padding (the length field may be longer than one byte), plus
  - 3 bytes to allow for the Le data object (if Le is present).
- The encrypted command data shall be preceded by the ISO/IEC 7816 padding indicator '01' and encapsulated in a BER-TLV data object with tag '86' and a length field coded according to ISO 8825-1;
- The Le byte, if present in the unprotected APDU, shall be contained in a BER-TLV data object with tag '96'.

The following diagram shows the message reformatting that is performed by the Off-Card Entity when a command is protected for confidentiality.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0278

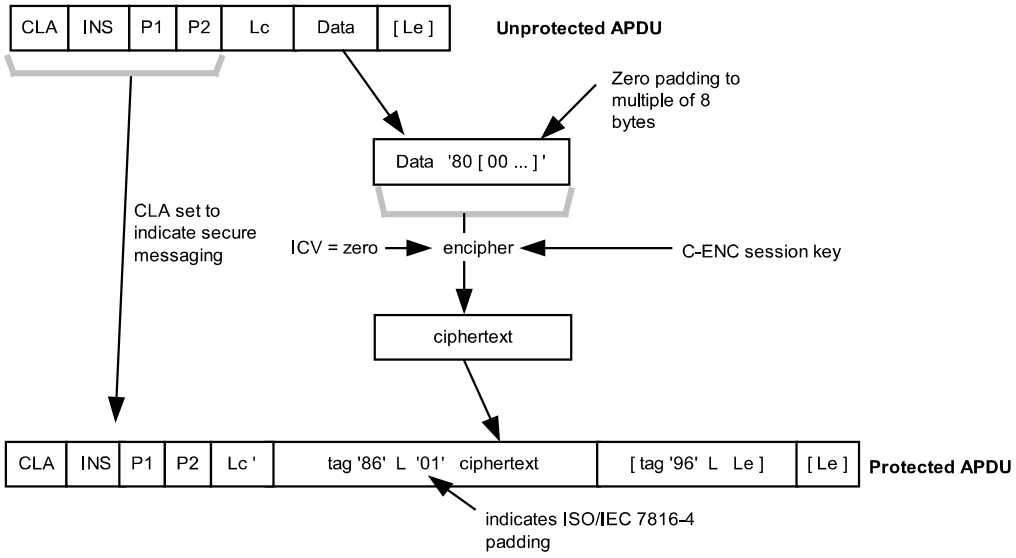


Figure F-11: Secure Messaging: Command message protected for confidentiality

F.3.2.3 APDU Command C-MAC and C-ENC Protection

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

No encryption shall be applied to a command where there is no command data field: in this case the message shall be protected as defined in section F.3.2.1 - *APDU Command C-MAC Protection*.

Otherwise the Off-Card Entity first encrypts the command data field of the command message being transmitted to the card as defined in section F.3.2.2.

A C-MAC is generated by an Off-Card Entity as defined in section F.3.2.1. Input data to the MAC calculation is first prepared as defined in ISO/IEC 7816-4:

- The following data is concatenated:
  - The command header CLA, INS, P1, P2 from the unprotected APDU, with the logical channel bits in the CLA byte set to zero, appended with four bytes '80 00 00 00';
  - If a command data field is present in the unprotected APDU, a BER-TLV data object with tag '87' containing the ISO/IEC 7816 padding indicator '01' followed by the encrypted command data;
  - If Le is present in the unprotected APDU, a BER-TLV data object with tag '97' containing the original Le value;
- DES padding is applied as defined in appendix B.4 - *DES Padding*.

To reflect the presence of a C-MAC and C-ENC protection of the command, the unprotected APDU shall be modified as follows:

- The class byte shall be modified to indicate that this APDU command includes secure messaging. This is achieved by setting to '11' bits 4-3 of a class byte indicating a logical channel number 0 to 4 (unprotected CLA set to '00' - '03' or '80' - '83') or by setting to '1' bit b6 of a class byte indicating a logical channel number 4 to 19 (unprotected CLA set to '40' - '4F' or 'C0' - 'CF'): see section 11.1.4. The logical channel bits are unchanged;.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0279

- Lc shall be incremented by:
  - 10 bytes to allow for the C-MAC data object, plus
  - 4 or more bytes to allow for the tag and length of the command data field data object, the padding indicator and the variable padding (the length field may be longer than one byte), plus
  - 3 bytes to allow for the Le data object (if Le is present).
- The encrypted command data shall be preceded by the ISO/IEC 7816 padding indicator '01' and encapsulated in a BER-TLV data object with tag '87' and a length field coded according to ISO 8825-1;
- The Le byte, if present in the unprotected APDU, shall be contained in a BER-TLV data object with tag '97';
- The C-MAC shall be encapsulated in a BER-TLV data object with tag '8E' and appended at the end of the command data field.

The following diagram shows the message reformatting that is performed by the Off-Card Entity when a command is protected for integrity and confidentiality.

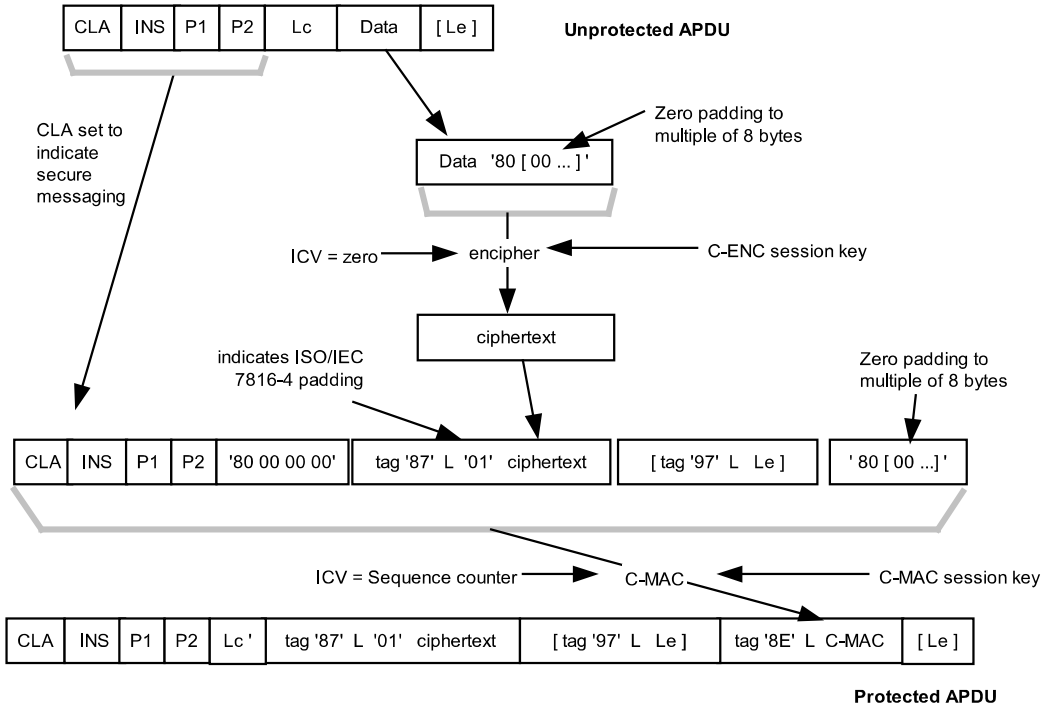


Figure F-12: Secure Messaging: Command message protected for integrity and confidentiality

F.3.2.4 APDU Response R-MAC Protection

This section applies where response integrity (R-MAC) is required but not confidentiality (R-ENC).

No R-MAC shall be generated and no protection shall be applied to a response where status bytes SW1 and SW2 indicate an error: in this case only status bytes shall be returned in the response.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0280

When R-MAC protection is required for a case 1 or case 3 command, the card shall process the command as a case 2 or case 4 command respectively and treat Le as if it were present and set to zero.

An R-MAC is generated by the card across the response data field (if present) and status bytes. Input data to the MAC calculation is first prepared as defined in ISO/IEC 7816-4:

- The following data is concatenated:
  - If a response data field is present in the unprotected APDU, a BER-TLV data object with tag '81' containing the complete original response data field, regardless of its contents and format;
  - A BER-TLV data object with tag '99', containing the original status word SW1 - SW2 value.
- DES padding is applied as defined in appendix B.4 - *DES Padding*.

An R-MAC is generated using the Secure Channel R-MAC session key, the encrypted sequence counter as the ICV as defined in section F.2.3, and the signature method described in appendix B.1.2.2 - *Single DES Plus Final Triple DES MAC* across the input data.

To reflect the presence of an R-MAC protection of the response, the unprotected APDU shall be modified as follows:

- The response data, if present in the unprotected APDU, shall be encapsulated in a BER-TLV data object with tag '81' and a length field coded according to ISO 8825-1;
- The status word SW1 - SW2 of the unprotected APDU shall be contained in a BER-TLV data object with tag '99';
- The R-MAC shall be encapsulated in a BER-TLV data object with tag '8E' and appended at the end of the response data field.

No padding is present in the transmitted APDU.

The Off-Card Entity, in order to verify the R-MAC, shall perform the same processing in order to generate an R-MAC and compare it with the transmitted R-MAC.

The following diagram shows the message reformatting that is performed by the card when a response message is protected for integrity.

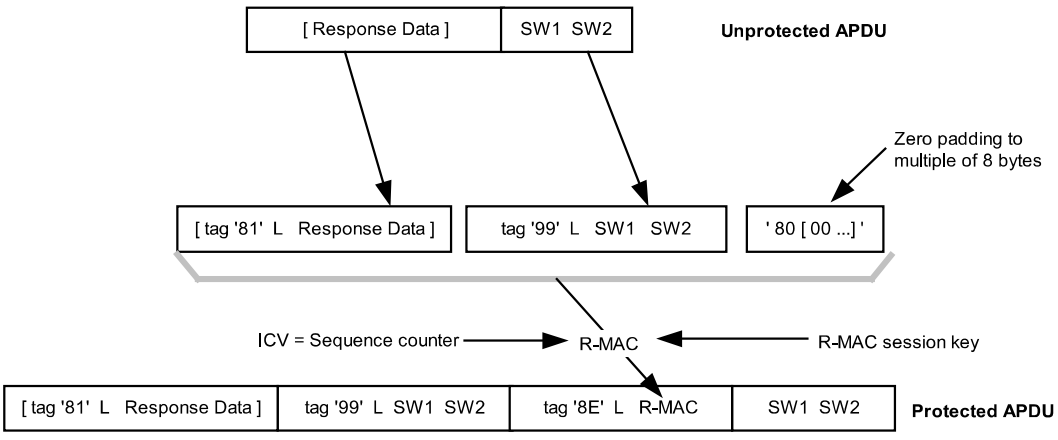


Figure F-13: Secure Messaging: Response message protected for integrity

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0281

F.3.2.5 APDU Response R-ENC Protection

This section applies where response confidentiality (R-ENC) is required but not integrity (R-MAC).

No protection shall be applied to a response where status bytes SW1 and SW2 indicate an error or where there is no response data field: in this case only status bytes shall be returned in the response.

Otherwise, the Security Domain encrypts the response data field. This includes any data within the data field that has already been protected for another purpose, such as secret or private keys encrypted with the data encryption session key.

Prior to encrypting the response data field, DES padding is applied as defined in appendix B.4 - *DES Padding*.

The padded response data field is then enciphered using triple DES in CBC mode as defined in appendix B.1.1.1 - *CBC Mode*, the R-ENC session key established during the Secure Channel initiation process and an ICV of zero.

To reflect the R-ENC protection of the response, the unprotected APDU shall be modified as follows:

- The encrypted response data shall be preceded by the ISO/IEC 7816 padding indicator '01' and encapsulated in a BER-TLV data object with tag '86' and a length field coded according to ISO 8825-1.

The following diagram shows the message reformatting that is performed by the card when a response message is protected for confidentiality.

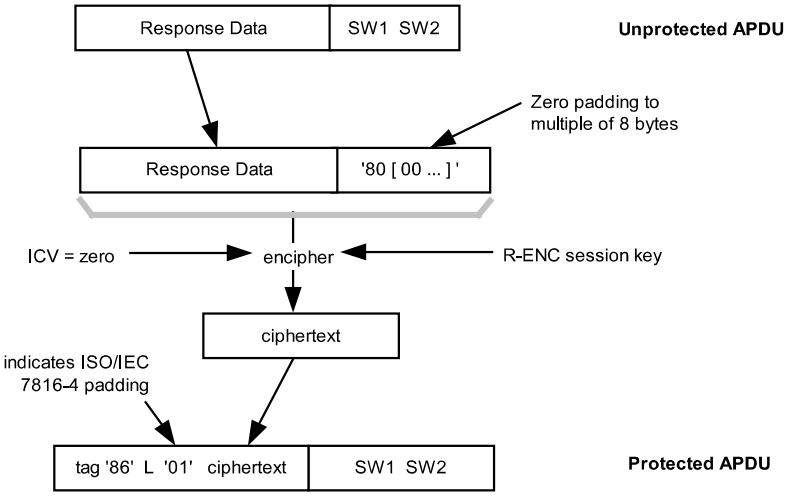


Figure F-14: Secure Messaging: Response message protected for confidentiality

F.3.2.6 APDU Response R-MAC and R-ENC Protection

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

No R-MAC or encryption shall be applied to a response where status bytes SW1 and SW2 indicate an error: in this case only status bytes shall be returned in the response.

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 F.3.2.4 - *APDU Response R-MAC Protection*.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0282

Otherwise, the card first encrypts the response data field of the response message being transmitted to the Off-Card Entity as defined in section F.3.2.5.

An R-MAC is then generated by the card as defined in section F.3.2.4. Input data to the MAC calculation is first prepared as defined in ISO/IEC 7816-4:

- The following data is concatenated:
  - If a response data field is present in the unprotected APDU, a BER-TLV data object with tag '87' containing the ISO/IEC 7816 padding indicator '01' followed by the encrypted response data;
  - A BER-TLV data object with tag '99' containing the original SW1 - SW2 status word value.
- DES padding is applied as defined in appendix B.4 - *DES Padding*.

To reflect the presence of an R-MAC and R-ENC protection of the response, the unprotected APDU shall be modified as follows:

- The encrypted response data shall be preceded by the ISO/IEC 7816 padding indicator '01' and encapsulated in a BER-TLV data object with tag '87' and a length field coded according to ISO 8825-1;
- The status word SW1 - SW2 of the unprotected APDU shall be contained in a BER-TLV data object with tag '99';
- The R-MAC shall be encapsulated in a BER-TLV data object with tag '8E' and appended at the end of the response data field.

No R-MAC padding is present in the transmitted APDU.

The Off-Card Entity, in order to verify the R-MAC, shall perform the same processing in order to generate an R-MAC and compare it with the transmitted R-MAC.

The following diagram shows the message reformatting that is performed by the card when a response message is protected for integrity and confidentiality.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0283

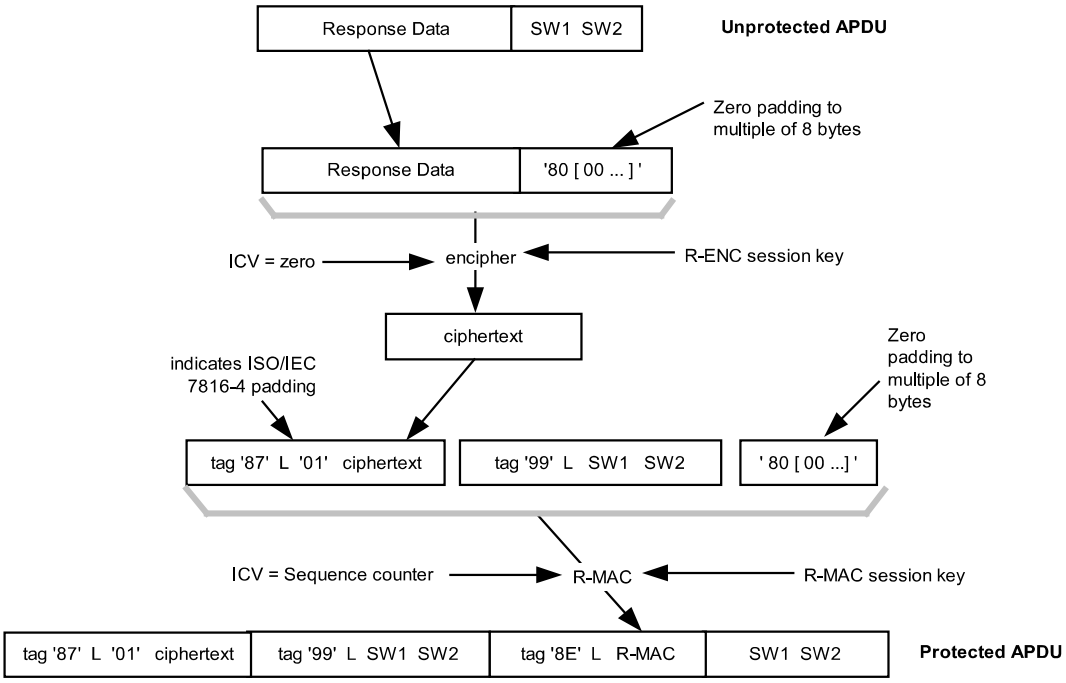


Figure F-15: Secure Messaging: Response message protected for integrity and confidentiality

F.3.2.7 Sensitive Data Encryption and Decryption

Data encryption is used when transmitting sensitive data to and from the card. For instance all keys transmitted to a card (e.g. in a PUT KEY command) should be encrypted. Data encryption is over and beyond the Current Security Level required for the Secure Channel Session. The encryption process uses the relevant data encryption session key (DEK) for sensitive data in command messages or for sensitive data in response messages. The encryption method uses DES in ECB or CBC mode depending on the Key Type of the DEK Key in the CRT: see appendix B.1.1.1 - *CBC Mode* or appendix B.1.1.2 - *ECB Mode*. If the key type is omitted for the DEK Key it shall be known implicitly. The sensitive data block length shall be constructed as a multiple of 8-byte long block before the encryption operations: the eventual padding method is application specific.

The encryption is performed across the sensitive data and the result of each encryption becomes part of the encrypted data. This encrypted data becomes part of the clear text data field in the command/response message. The decryption is the exact opposite of the above operation: in particular, no padding is removed by the decryption operation.

F.4 Commands

Because certificates, digital signatures and some data fields can be long, command and response chaining as defined in ISO/IEC 7816-4 is used to transfer successive data blocks.

With command chaining, the command data is sent in multiple APDUs, the command data being segmented arbitrarily. All except the final command in the chain shall indicate command chaining by setting to '1' bit 5 of the class byte according to ISO/IEC 7816-4.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0284

With response chaining, the response data is sent in multiple APDUs, the response data being segmented arbitrarily.

Command	Secure Channel Initiation	
	signature without message recovery	signature with message recovery
EXTERNAL AUTHENTICATE	✓	✓
GET CHALLENGE	✓	✓
GET DATA [certificate]	✓	✓
INTERNAL AUTHENTICATE	✓	✓
MANAGE SECURITY ENVIRONMENT	✓	✓
PERFORM SECURITY OPERATION [decipher]	✓	
PERFORM SECURITY OPERATION [verify certificate]	✓	✓

Table F-13: SCP10 Command Support

Table F-14 summarizes the minimum security requirements for the APDU commands.

Command	Minimum Security
EXTERNAL AUTHENTICATE	Validated PK.OCE.AUT and card challenge
GET CHALLENGE	None
GET DATA [certificate]	None
INTERNAL AUTHENTICATE	Current Security Level is at least AUTHENTICATED or ANY_AUTHENTICATED
MANAGE SECURITY ENVIRONMENT	None
PERFORM SECURITY OPERATION [decipher]	None
PERFORM SECURITY OPERATION [verify certificate]	None

Table F-14: Minimum Security Requirements for SCP10 commands

The following table provides the list of SCP10 command support per card Life Cycle State.

Command	OP_READY			INITIALIZED			SECURED			CARD_LOCKED		TERMINATED	
	AM SD	DM SD	SD	AM SD	DM SD	SD	AM SD	DM SD	SD	FA SD	SD	FA SD	SD
EXTERNAL AUTHENTICATE	✓	✓		✓	✓		✓	✓		✓			
GET CHALLENGE	✓	✓		✓	✓		✓	✓		✓			

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0285

Command	OP_READY			INITIALIZED			SECURED			CARD_LOCKED		TERMINATED	
	AM SD	DM SD	SD	AM SD	DM SD	SD	AM SD	DM SD	SD	FA SD	SD	FA SD	SD
GET DATA [certificate]	✓	✓		✓	✓		✓	✓		✓			
INTERNAL AUTHENTICATE	✓	✓		✓	✓		✓	✓		✓			
MANAGE SECURITY ENVIRONMENT	✓	✓		✓	✓		✓	✓		✓			
PERFORM SECURITY OPERATION [decipher]	✓	✓		✓	✓		✓	✓		✓			
PERFORM SECURITY OPERATION [verify certificate]	✓	✓		✓	✓		✓	✓		✓			

Table F-15: SCP10 command support per Card Life Cycle State

Legend of Table F-13 and Table F-15:

AM SD: Security Domain with Authorized Management privilege.

DM SD: Supplementary Security Domain with Delegated Management privilege.

FA SD: Security Domain with Final Application privilege. (Note: command support for an Application with Final Application Privilege which is not a Security Domain is subject to Issuer policy, within the constraints of what is permitted according to the card Life Cycle State.)

SD: Other Security Domain.

✓ : Support required.

Blank cell: Support optional.

Striped cell: Support prohibited.

F.4.1 EXTERNAL AUTHENTICATE Command

F.4.1.1 Definition and Scope

This command is used to authenticate the Off-Card Entity by the Security Domain. This command is also used with the key agreement option to support session key establishment. It shall be immediately preceded (on the same logical channel of the same card I/O interface) by a GET CHALLENGE command. It may be followed (on the same logical channel of the same card I/O interface) by an INTERNAL AUTHENTICATE command.

F.4.1.2 Command Message

The EXTERNAL AUTHENTICATE command message is coded as follows:

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0286

Code	Value	Meaning
CLA	'00' - '03', '40' - '4F', '10' - '13' or '50' - '5F'	See section 11.1.4
INS	'82'	EXTERNAL AUTHENTICATE
P1	'00'	Reference control parameter P1: no information given
P2	'00'	Reference control parameter P2: no information given
Lc	'xx'	Length of Entity Signature (key transport) or encrypted Off-Card Entity Signature (key agreement)
Data	'xx xx…'	Command data field
Le	-	Not present

Table F-16: EXTERNAL AUTHENTICATE Command Message

A Security Domain may support other values of Reference Control Parameters P1 and P2 as defined in ISO/IEC 7816-4.

F.4.1.3 Data Field Sent in the Command Message

The data field of the EXTERNAL AUTHENTICATE command message contains the Off-Card Entity Signature (key transport) or encrypted Off-Card Entity Signature (key agreement) - see section F.1.3.4 - *Off-Card Entity Authentication* for further details.

F.4.1.4 Data Field Returned in the Response Message

No data is returned by this command.

F.4.1.5 Processing State Returned in the Response Message

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

The command may either return a general error condition as listed in section 11.1.3 - *General Error Conditions* or one of the following specific error and warning conditions.

SW1	SW2	Meaning
'63'	'00'	Verification of certificate failed
'94'	'84'	Algorithm not supported

Table F-17: Error Conditions

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0287

F.4.2 GET CHALLENGE Command

F.4.2.1 Definition and Scope

This command is used to obtain a random challenge from the Security Domain, to support authentication of the Off-Card Entity to the Security Domain. It precedes the EXTERNAL AUTHENTICATE command. It shall have been preceded (on the same logical channel of the same card I/O interface), immediately or not, by a MANAGE SECURITY ENVIRONMENT or PERFORM SECURITY OPERATION [verify certificate] command.

F.4.2.2 Command Message

The GET CHALLENGE command message is coded according to the following table:

Code	Value	Meaning
CLA	'00' - '03' or '40' - '4F'	See section 11.1.4
INS	'84'	GET CHALLENGE
P1	'00'	Reference Control Parameter P1: no information given
P2	'00'	Reference Control Parameter P2: no information given
Lc	Absent	
Data	Absent	
Le	'00'	

Table F-18: GET CHALLENGE Command Message

A Security Domain may support other values of Reference Control Parameters P1 and P2 as defined in ISO/IEC 7816-4.

F.4.2.3 Data Field Sent in the Command Message

There is no command data.

F.4.2.4 Data Field Returned in the Response Message

Data returned comprises a card challenge, coded on 8 bytes with the key agreement option and 16 bytes with the key transport option.

F.4.2.5 Processing State Returned in the Response Message

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

The command may return a general error condition as listed in section 11.1.3 - *General Error Conditions*.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0288

F.4.3 GET DATA [certificate] Command

F.4.3.1 Definition and Scope

The GET DATA [certificate] command is used to retrieve either information about all certificates that can be retrieved from the card, or a single certificate. This command may be issued at any time, in particular it may be interleaved with PERFORM SECURITY OPERATION [verify certificate] commands. If it is issued when a Secure Channel Session is active, it must comply with the Current Security Level of that Secure Channel Session.

F.4.3.2 Command Message

The following command is used to obtain certificate information or a single certificate from the Security Domain.

Code	Value	Meaning
CLA	'00' - '0F', '40' - '4F', '60' - '6F', '80' - '8F', 'C0' - 'CF' or 'E0' - 'EF'	See section 11.1.4
INS	'CA' or 'CB'	If CLA = '00' - '0F', '40' - '4F' or '60' - '6F', even or odd instruction code 'CA' or 'CB' If CLA = '80' - '8F', 'C0' - 'CF' or 'E0' - 'EF', even instruction code 'CA'
P1 P2		Reference Control Parameters P1 and P2: there are three options:
	'7F 21'	Tag of certificate
	'50 31'	Data object identifier of EF.OD
	'xx xx'	(Any other value) data object identifier of a certificate
Lc	'xx' or omitted	Not present if P1 P2 = '7F 21', otherwise length of command data
Data	'xx xx...' or omitted	Not present, or command data
Le	'00'	

Table F-19: GET DATA [certificate] Command Message

F.4.3.3 Reference Control Parameters P1 and P2

If P1 P2 = '7F 21', the command is a request to retrieve the certificate of the Security Domain's default public key, CERT.SD.AUT.

Otherwise, for any value other than those defined in section 11.3 - *GET DATA Command*, the command is a request to access EF.OD or another certificate and the instruction code shall be set to 'CA' if the class byte indicates a GlobalPlatform command (CLA set to '80' - '8F', 'C0' - 'CF' or 'E0' - 'EF') or 'CB' if the class byte indicates an ISO command (CLA set to '00' - '0F', '40' - '4F' or '60' - '6F').

- If P1 P2 = '50 31', the command is a request to retrieve details of all available certificates held by the Security Domain for retrieval and verification by an Off-Card Entity;

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0289

- Otherwise, for any other value of P1 P2, the command shall be treated as a request to retrieve a certificate whose pointer is given in P1 P2. This would typically follow a command with P1 P2 = '50 31', where the Cryptographic Information Objects returned in the response have pointers to individual certificates. However, the Off-Card Entity may already know the location of a required certificate, and issue this command directly.

F.4.3.4 Data Field Sent in the Command Message

When P1-P2 is different from '7F21', the command data shall be present and coded as follows:

Name	Length	Name	Presence
Tag list tag	1	'5C'	Mandatory
Tag list length	1	'00' (empty, indicating 'retrieve all data')	Mandatory

Table F-20: GET DATA [certificate] Command Data Message

F.4.3.5 Data Field Returned in the Response Message

When the command is issued to retrieve a certificate (P1-P2 different from '5031') and the class byte indicates a GlobalPlatform command (CLA set to '80' - '8F', 'C0' - 'CF' or 'E0' - 'EF'), the certificate shall be returned TLV-coded as follows:

Name	Length	Name	Presence
Certificate tag	2	'7F21'	Conditional
Certificate length	1, 2 or 3	'00' - '7F', or '81 80' - '81 FF' or '82 01 00' - '82 FF FF'	Conditional
Certificate data	n	'xxxx...' (as defined in section F.1.2.4).	Mandatory

Table F-21: GET DATA [certificate] Response Data Field - Certificate

When the command is issued to retrieve a certificate (P1-P2 different from '5031') and the class byte indicates an ISO command (CLA set to '00' - '0F', '40' - '4F' or '60' - '6F'), only the certificate value shall be returned.

If the command was issued to retrieve certificate information details (P1-P2 = '5031'), the contents of EF.OD listing all the Cryptographic Information Objects for the different certificates held in by the Security Domain shall be returned as defined in section F.1.2.3.

F.4.3.6 Processing State Returned in the Response Message

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

The command may either return a general error condition as listed in section 11.1.3 - *General Error Conditions* or one of the following specific error and warning conditions:

SW1	SW2	Meaning
'6A'	'80'	Incorrect values in command data
'6A'	'88'	Referenced data not found

Table F-22: Error Conditions

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0290

F.4.4 INTERNAL AUTHENTICATE Command

F.4.4.1 Definition and Scope

This command is used to authenticate the Security Domain, by the Off-Card Entity. This command is also used with the key agreement option, to support session key establishment. It shall be immediately preceded (on the same logical channel of the same card I/O interface) by an EXTERNAL AUTHENTICATE command.

F.4.4.2 Command Message

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

Code	Value	Meaning
CLA	'00' - '03' or '40' - '4F'	See section 11.1.4
INS	'88'	INTERNAL AUTHENTICATE
P1	'00'	Reference control parameter P1: no information given
P2	'00'	Reference control parameter P2: no information given
Lc	'xx'	Length of Off-Card Entity challenge
Data	'xx xx…'	Command data field
Le	'00'	

Table F-23: INTERNAL AUTHENTICATE Command Message

A Security Domain may support other values of Reference Control Parameters P1 and P2 as defined in ISO/IEC 7816-4.

F.4.4.3 Data Field Sent in the Command Message

The data field of the INTERNAL AUTHENTICATE command message contains the following data:

Name	Length	Name	Presence
Off-Card Entity challenge	8 or 16	'xxxx...'	Mandatory
Off-Card Entity id	8	'xxxx...' (part of CERT.OCE.AUT)	Mandatory

Table F-24: INTERNAL AUTHENTICATE Command Data Field

Off-Card Entity challenge is 16 bytes for the key transport option, 8 bytes for the key agreement option.

F.4.4.4 Data Field Returned in the Response Message

The Card Signature (key transport) or encrypted Card Signature (key agreement) is returned - see section F.1.3.3 for details.

F.4.4.5 Processing State Returned in the Response Message

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

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0291

The command may either return a general error condition as listed in section 11.1.3 - *General Error Conditions* or one of the following specific error and warning conditions:

SW1	SW2	Meaning
'61'	'xx'	Response data incomplete, 'xx' more bytes available

Table F-25: Warning Conditions

SW1	SW2	Meaning
'6A'	'80'	Incorrect values in command data

Table F-26: Error Conditions

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0292

F.4.5 MANAGE SECURITY ENVIRONMENT Command

F.4.5.1 Definition and Scope

This command selects the Secure Channel Protocol '10' and its options as well as defining specific keys to be used by the Security Domain. If a Secure Channel Session is active on this logical channel and card I/O interface, it shall be terminated on receipt of this command, regardless of the validity of the command.

F.4.5.2 Command Message

The MANAGE SECURITY ENVIRONMENT command message is coded according to the following table:

Code	Value	Meaning
CLA	'00' - '03' or '40' - '4F'	See section 11.1.4
INS	'22'	MANAGE SECURITY ENVIRONMENT
P1	'81' or 'C1'	Reference Control Parameter P1 '81': External (Off-Card Entity) Authentication only 'C1': External and Internal (Mutual) Authentication
P2	'A4' or 'B6'	Reference Control Parameter P2: 'A4': Authentication: no certificate verification will be performed by the card 'B6': Digital signature: certificate verification will be performed by the card
Lc	'xx'	Length of command data
Data	'xx xx…'	Off-Card Entity data
Le	-	Not present

Table F-27: MANAGE SECURITY ENVIRONMENT Command Message

F.4.5.3 Reference Control Parameter P1

The value of P1 is based on ISO/IEC 7816-4, as follows:

b8	b7	b6	b5	b4	b3	b2	b1	Description
1	-	-	-	-	-	-	-	Verification, encipherment, external authentication and key agreement
-	1	-	-	-	-	-	-	Computation, decipherment, internal authentication and key agreement
-	-	-	-	-	-	-	1	SET
-	-	X	X	X	X	X	-	Values defined in ISO/IEC 7816-4

Table F-28: MANAGE SECURITY ENVIRONMENT Reference Control Parameter P1

F.4.5.4 Reference Control Parameter P2

This is set according to the template that is appropriate for the subsequent message flow, as defined in ISO/IEC 7816-4: 'A4' if certificate verification by the card is omitted, 'B6' if certificate verification is to be performed by the card.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0293

F.4.5.5 Data Field Sent in the Command Message

The command data field is formatted as follows:

Tag	Length	Name	Presence
'80'	2	Cryptographic mechanism reference: SCP id + options 'i' (= scp || i)	Mandatory
'83'	0 or 1-n	Public key reference	Conditional
'84'	0 or 1-n	Private key reference	Conditional

Table F-29: MANAGE SECURITY ENVIRONMENT Command Data Field

The 'cryptographic mechanism reference' (tag '80', value '10') designates GlobalPlatform asymmetric Secure Channel Protocol '10' (SCP10) and its options, and distinguishes it from any other protocol that might be supported by the selected Issuer Security Domain, Security Domain or Application. Option 'i' is described in section F.1.1.

The 'public key reference' (tag '83') designates the public key to be used by the Security Domain in subsequent cryptographic operations during Secure Channel Session initiation. The referenced public key shall have already been validated by the Security Domain or shall be the public key of the Security Domain's Trust Point for External Authentication. If no reference value is provided in the command, the Security Domain shall use by default the public key of its Trust Point for External Authentication, designated PK.TP_EX.AUT, as the first key to use for certificate verification within a session: see section F.1.2.1.

The 'private key reference' (tag '84') designates the private key to be used by the Security Domain in subsequent cryptographic operations during Secure Channel Session initiation. The referenced private key shall be present and known to the Security Domain. If no reference value is provided in the command, the Security Domain shall use by default its private key designated SK.SD.AUT

A Security Domain may support other data elements as defined in ISO/IEC 7816-4.

F.4.5.6 Data Field Returned in the Response Message

There is no data returned from this command.

F.4.5.7 Processing State Returned in the Response Message

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

The command may either return a general error condition as listed in section 11.1.3 - *General Error Conditions* or one of the following specific error and warning conditions.

SW1	SW2	Meaning
6A	88	Referenced data not found
94	84	Algorithm not supported

Table F-30: Error Conditions

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0294

F.4.6 PERFORM SECURITY OPERATION [decipher] Command

F.4.6.1 Definition and Scope

This command is used with the session key transport option, and transmits the DES session keys from the Off-Card Entity to the Security Domain. It shall have been preceded (on the same logical channel of the same card I/O interface), immediately or not, by a MANAGE SECURITY ENVIRONMENT or PERFORM SECURITY OPERATION [verify certificate] command.

F.4.6.2 Command Message

The PERFORM SECURITY OPERATION [decipher] command message is coded according to the following table:

Code	Value	Meaning
CLA	'00' - '03', '40' - '4F', '10' - '13' or '50' - '5F'	See section 11.1.4
INS	'2A'	PERFORM SECURITY OPERATION [decipher]
P1	'80'	Reference Control Parameter P1: clear text object
P2	'84'	Reference Control Parameter P2: cryptogram (plain value encoded in BER-TLV) present in the command
Lc	'xx'	Length of encrypted data
Data	'xx xx…'	Encrypted data
Le	-	Not present

Table F-31: PERFORM SECURITY OPERATION [decipher] Command Message

A Security Domain may support other values of Reference Control Parameters P1 and P2 as defined in ISO/IEC 7816-4.

F.4.6.3 Data Field Sent in the Command Message

The data field of the PERFORM SECURITY OPERATION [decipher] command message contains the Encrypted Off-Card Entity Session Data.

The Off-Card Entity session keys are in control reference templates as defined in section F.3.1.2, one template per session key, concatenated for encryption. Off-Card Entity Session Key Data is formed by padding the session key CRTs to the length of the Security Domain public key modulus as shown in the table below, and encrypting the result with the Security Domain public key (PK.SD.AUT).

Meaning	Length	Meaning	Presence
Padding	2	'0002'	
Padding	8-n	'FF'...'FF'	
Padding	1	'00'	
Tag of Security Level ('D3')	1	'D3'	Mandatory

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0295

Meaning	Length	Meaning	Presence
Length of Security Level	1	‘01’	Mandatory
Security Level	1	‘xx’	Mandatory
CRT tag	1	‘B4’ (CCT) or ‘B8’ (CT)	Mandatory
Length of CRT	1	'00' - '7F'	Mandatory
CRT contents (with session key)	n	'xxxx...'	Mandatory
…..	….	…..	…
CRT tag	1	‘B4’ (CCT) or ‘B8’ (CT)	Optional
Length of CRT	1	‘00’ - ‘7F’	Conditional
CRT contents (with session key)	n	'xxxx...'	Conditional

Table F-32: Off-Card Entity Session Key Data - Clear Text before Encryption

F.4.6.4 Data Field Returned in the Response Message

No data is returned from this command.

F.4.6.5 Processing State Returned in the Response Message

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

The command may either return a general error condition as listed in section 11.1.3 - *General Error Conditions* or one of the following specific error and warning conditions.

SW1	SW2	Meaning
'6A'	'80'	Incorrect values in command data

Table F-33: Error Conditions

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0296

F.4.7 PERFORM SECURITY OPERATION [verify certificate] Command

F.4.7.1 Definition and Scope

This command is used to provide a certificate to the Security Domain for verification. It may be preceded (on the same logical channel of the same card I/O interface) by a MANAGE SECURITY ENVIRONMENT command or a PERFORM SECURITY OPERATION [verify certificate] command and may be interleaved with GET DATA [certificate] commands.

F.4.7.2 Command Message

The PERFORM SECURITY OPERATION [verify certificate] command message is coded according to the following table:

Code	Value	Meaning
CLA	'00' - '03', '40' - '4F', '10' - '13' or '50' - '5F'	See section 11.1.4
INS	'2A'	PERFORM SECURITY OPERATION [verify certificate]
P1	'00'	Reference Control Parameter P1: no object in the response
P2	'AE' or 'BE'	Reference Control Parameter P2: input template for certificate verification present in the command 'AE': non self descriptive card verifiable certificate (only the concatenated value fields are certified) 'BE': self descriptive card verifiable certificate (the TLV data elements are certified)
Lc	'xx'	Length of certificate data
Data	'xx xx…'	Certificate data
Le	-	Not present

Table F-34: PERFORM SECURITY OPERATION [verify certificate] Command Message

F.4.7.3 Data Field Sent in the Command Message

The command data field is certificate data as follows.

Name	Length	Name	Presence
Certificate tag	2	'7F21'	Mandatory
Certificate length	1, 2 or 3	'00' - '7F', or '81 80' - '81 FF' or '82 01 00' - '82 FF FF'	Mandatory
Certificate data	n	'xxxx...' (as described in section F.1.2.4)	Mandatory

Table F-35: PERFORM SECURITY OPERATION [verify certificate] Command Data Field

The Security Domain verifies the certificate presented using the Current Public Key as known to the Security Domain. For the first certificate presented in the session, the Current Public Key is either the default Public Key - the Public Key of the Trust Point, or another Public Key as announced in the MANAGE SECURITY ENVIRONMENT command. A series of certificates can be presented to the Security Domain in subsequent PERFORM SECURITY OPERATION [verify certificate] commands. For

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0297

each subsequent PERFORM SECURITY OPERATION [verify certificate] command, the Current Public Key is the one that was certified in the certificate presented in the previous PERFORM SECURITY OPERATION [verify certificate] command.

F.4.7.4 Data Field Returned in the Response Message

No data is returned from this command.

F.4.7.5 Processing State Returned in the Response Message

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

The command may either return a general error condition as listed in section 11.1.3 - *General Error Conditions* or one of the following specific error and warning conditions.

SW1	SW2	Meaning
'63'	'00'	Verification of the certificate failed
'68'	'83'	The last command of the chain was expected
'6A'	'80'	Incorrect values in command data

Table F-36: Error Conditions

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0298

G Trusted Framework Inter-Application Communication

GlobalPlatform supports a general Trusted Framework scheme for inter-application communication. This general scheme is compatible with the CAT Runtime Environment and secure communication as defined in ETSI TS 102 225, TS 102 241 and related specifications. In summary:

- An Application, the Receiving Entity, receives a message from an off-card entity, whose contents is destined to another Application (the Target Application);
- The Receiving Entity interacts with a Trusted Framework on the card to communicate with the Target Application;
- The Trusted Framework handles the security of the inter-application communication by applying its own rules to the interaction, which may include finding the Security Domain associated with the Target Application and having it handle the security of the incoming message;
- The Trusted Framework finds the appropriate interface of the Target Application and passes to it the incoming message contents;
- The Target Application processes the event and eventually provides to the Trusted Framework some response message to be returned;
- The Trusted Framework, being responsible of the inter-application communication security, applies its own security rules to the response message, which may include requesting the Security Domain associated with the Target Application to handle the response message security;
- The Receiving Entity handles the transmission of the response message to the off-card entity.

Passing the incoming message through to the Target Application, and returning the Target Application's response to the off-card entity implies security requirements for not only the Trusted Framework but also the Receiving Entity. Each Application present on the card playing the role of a Receiving Entity shall:

- Enforce the Issuer's security rules for inter-application communication;
- Ensure that incoming messages are properly provided unaltered to the Trusted Framework;
- Ensure that any response messages are properly returned unaltered to the off-card entity.

The 'Trusted Path' privilege qualifies an Application as a Receiving Entity.

To be able to interact with a required Trusted Framework, the Receiving Entity shall obtain a handle to the required Trusted Framework.

To be able to receive incoming messages from a Receiving Entity, the Target Application declares its interface(s) to the corresponding Trusted Framework. The Target Application is registered at installation time for each event it is capable of handling.

When requested to perform inter-application communication, the Trusted Framework shall check that

- The Receiving Entity has the 'Trusted Path' privilege;
- The Target Application is registered as being willing to accept the event or incoming message.

Trusted Frameworks shall comply with the general requirements stated in this section but are otherwise outside the scope of this version of the specification.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0299

H GlobalPlatform Data Values and Card Recognition Data

H.1 Data Values

The Object Identifier (OID) assigned to GlobalPlatform is as follows:

GlobalPlatform OID ::= `{iso(1) member-body(2) country-USA(840) GlobalPlatform(114283)}`

The hexadecimal representation of the above OID is '2A864886FC6B'. The Object Identifier value for Card Recognition Data `{globalPlatform 1}` or `{1 2 840 114283 1}`, which also identifies GlobalPlatform as the Tag Allocation Authority for Card Recognition Data objects has an hexadecimal representation of '2A864886FC6B01' and so on and so forth for all subsequent GlobalPlatform Object Identifiers.

The Registered Application Provider Identifier (RID) assigned to GlobalPlatform is as follows:

GlobalPlatform RID = 'A000000151'

The default AID of the Issuer Security Domain, based on the above ISO assigned RID, is 'A0000001510000'.

The encoding of an OID is defined in ISO/IEC 8825-1. The definition of a RID is described in ISO/IEC 7816-4.

H.2 Structure of Card Recognition Data

All data is TLV encoded. Application tags not defined in this specification are RFU. GlobalPlatform may assign additional Application tags in the future.

Many of the data objects contain Object Identifiers (OIDs). An OID is made up of a series of numeric values. Each OID is shown here in curly brackets, with its component values separated by spaces. The actual numeric values of some of the symbolic OID values shown, such as “globalPlatform”, are defined in this document and may themselves comprise a series of values.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0300

Tag	Explanation	Length	Value	Presence
'66'	Tag for 'Card Data'	variable - see note 1	Data objects identified in 7816-6, including tag '73'	Mandatory
'73'	Tag for 'Card Recognition Data'	variable	Data objects listed below	Mandatory
'06'	Universal tag for "Object Identifier" (OID)	variable	{globalPlatform 1} OID for Card Recognition Data, also identifies Global Platform as the Tag Allocation Authority	Mandatory
'60' '06'	Application tag 0 'OID' tag	variable variable	{globalPlatform 2 v} OID for Card Management Type and Version - see note 2	Mandatory
'63' '06'	Application tag 3 'OID' tag	variable variable	{globalPlatform 3} OID for Card Identification Scheme - see note 3	Mandatory
'64' '06'	Application tag 4 'OID' tag	variable variable	{globalPlatform 4 scp i} OID for Secure Channel Protocol of the Issuer Security Domain and its implementation options- see note 4	Mandatory
'65'	Application tag 5	variable	Card configuration details - see note 5	Optional
'66'	Application tag 6	variable	Card / chip details - see note 6	Optional
'67'	Application tag 7	variable	Issuer Security Domain's Trust Point certificate information - see note 7	Optional
'68'	Application tag 8	variable	Issuer Security Domain certificate information - see note 8	Conditional

Table H-1: Structure of Card Recognition Data

Note 1: Card Data should not exceed 127 bytes. This recommendation is to avoid the complexity of extending the length field of the '66' and '73' data objects and also to minimize the overheads involved in transferring the data.

Note 2: Tag '60': The OID **{globalPlatform 2 v}** identifies a card that conforms to the GlobalPlatform Card Specification version "**v**". Thus a card conforming to the GlobalPlatform Card Specification 2.2 would use OID **{globalPlatform 2 2 2}** and a card conforming to the GlobalPlatform Card Specification 2.1.1 would use OID **{globalPlatform 2 2 1 1}**.

Note 3: Tag '63': The OID **{globalPlatform 3}** indicates a GlobalPlatform card that is uniquely identified by the Issuer Identification Number (IIN) and Card Image Number (CIN), as defined in sections 7.4.1.1 - *Issuer Identification Number* and 7.4.1.2 - *Card Image Number*. The objective is that an off-card entity is able to construct a globally unique identifier for the card by concatenating this **{globalPlatform 3}** OID, the IIN and the CIN.

Note 4: Tag '64'. The OID **{globalPlatform 4 scp i}** identifies the Secure Channel Protocol of the Issuer Security Domain. "**scp**" identifies the Secure Channel Protocol identifier as defined in section 10.7 - *Secure Channel Protocol Identifier*. "**i**" identifies the eventual implementation options as defined in appendix D.1.1 - *SCP01 Secure Channel* for SCP01, appendix E.1.1 - *SCP02 Secure Channel* for SCP02 or appendix F.1.1 - *SCP10 Secure Channel* for SCP10.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0301

Note 5: The data object with tag '65' may contain information about the GlobalPlatform implementation details or commonly used Card Issuer options. Such information shall be TLV encoded. The structure of this data object is under definition by GlobalPlatform.

Note 6: Tag '66': this data object may contain information about the card and chip implementation, such as the operating system/runtime environment or a security kernel. Such information shall be TLV encoded and may consist of one (or more) OID(s), each OID being introduced by tag '06' and indicating the organization responsible for specifying the operating system, runtime environment or security kernel, and the identification of the corresponding specification and its version number.

Note 7: Tag '67': this may contain information on the certification policies, certificate formats and certificate ids associated with the Issuer Security Domain's Trust Point (TP_ISD), primarily relating to the use of a public key Secure Channel Protocol. Such information shall be TLV encoded and may consist of one (or more) OID(s).

Note 8: Tag '68': this may contain information such as certificate types, formats and ids associated with on-card Security Domains relating to the use of a public key Secure Channel Protocol. Such information shall be TLV encoded and may consist of one (or more) OID(s) .

H.3 Security Domain Management Data

The Security Domain management data may be returned in the SELECT response message within template '73' as described in section 11.9.3.1 - *Data Field Returned in the Response Message.* When present, the Security Domain management data shall be coded as follows.

Tag	Explanation	Length	Value	Presence
'73'	Template	variable	Data objects listed below	Mandatory
'06'	Universal tag for "Object Identifier" (OID)	variable	Identifies Global Platform as the Tag Allocation Authority	Mandatory
'60' '06'	Application tag 0 'OID' tag	variable variable	{globalPlatform 2 v} OID for Card Management Type and Version	Optional
'63' '06'	Application tag 3 'OID' tag	variable variable	{globalPlatform 3} OID for Card Identification Scheme	Optional
'64' '06'	Application tag 4 'OID' tag	variable variable	{globalPlatform 4 scp i} OID for Secure Channel Protocol of the selected Security Domain and its implementation options	Optional
'65'	Application tag 5	variable	Card configuration details	Optional
'66'	Application tag 6	variable	Card / chip details	Optional
'67'	Application tag 7	variable	Security Domain's Trust Point certificate information	Optional
'68'	Application tag 8	variable	Security Domain certificate information	Conditional

Table H-2: Security Domain Management Data

The data objects in this table are equivalent to those in Table H-1, but are specific to this Security Domain and override data objects in Card Recognition Data. See Table H-1 and the notes that follow it for further details of each data object.

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0302

END OF DOCUMENT

Copyright © 2006-2011 GlobalPlatform Inc. All Rights Reserved.
The technology provided or described herein is subject to updates, revisions, and extensions by GlobalPlatform. Use of this information is governed by the GlobalPlatform license agreement and any use inconsistent with that agreement is strictly prohibited.

Petitioner Samsung Ex-1006, 0303