Return to Menu

MasterCard International

Information about this Update

Update

This is the January 2005 update to your *Public Key Infrastructure (PKI) — Certification Authority Interface Specification.*

Billing

We will bill you for this document in printed format. Please refer to the *MasterCard Consolidated Billing System Manual* for billing-related information.

Questions?

If you have questions about this manual, please contact the Customer Operations Services team or your regional help desk. Please refer to "Using this Manual" for more contact information.

MasterCard is Listening…

Please take a moment to provide us with your feedback about the material and usefulness of the *Public Key Infrastructure (PKI) — Certification Authority Interface Specification* using the following e-mail address:

doc@mastercard.com

We continually strive to improve our publications. Your input will help us accomplish our goal of providing you with the information you need.

MasterCard International

Summary of Changes

***Public Key Infrastructure (PKI) — Certification Authority Interface Specification*, January 2005**

Description of Change	Where to Look
Updates to reflect the changes in the Public Key Certification Service and the new communication mechanism put in place between MasterCard and its members.	Whole manual

MasterCard International

Public Key Infrastructure (PKI) — Certification Authority Interface Specification

To Support M/Chip

January 2005

Media

This document is available:

- On MasterCard OnLine®
- On the *MasterCard Electronic Library (CD-ROM)*
- In printed format
- On your PC after installation of the PKI Member Package CD-ROM.

Manuals and Publications
Chaussée de Tervuren 198A
B-1410 Waterloo
Belgium
Fax: + 32 2 352 5353

e-mail: doc@mastercard.com

Using this Manual

Chapter 1 Interface Specification

Chapter 2 High-level Data Formats

Chapter 3 Certificates and Associated Data

Chapter 4 Self-signed Payment System Public Keys

Chapter 5 Self-signed Issuer Public Keys

Chapter 6 Issuer Public Key Certificates

Chapter 7 Hash Codes for Transferred Public Keys

Chapter 8 File Naming Conventions

Using this Manual

This chapter contains information that helps you understand and use this manual.

Purpose

This manual specifies the data structures and formats that members (issuers and/or acquirers) will use to exchange asymmetric key data with the MasterCard Public Key Certification Service.

Wherever possible, the 'EMV' certificate formats are used (refer to *EMV2000: Integrated Circuit Card Specification for Payment Systems – Book 2: Security & Key Management* for details).

Audience

This manual is intended for all member and vendor security and operational personnel involved in the public key certification process and, as such, with the MasterCard Certification Authority.

Overview

The following table provides an overview of this manual:

Chapter	Description
Table of Contents	A list of the manual's chapters and sections. Each entry references a chapter and page number.
Using this Manual	A description of the manual's purpose and its contents.
1 Interface Specification	Provides an overview of communications and data formats in the key transfer interface.
2 High-level Data Formats	Describes the main data formats transferred using the key transfer interface.
3 Certificates and Associated Data	Provides an overview of how certificates are constructed and verified, and a detailed description of their content.
4 Self-signed Payment System Public Keys	Describes the self-signed Payment System Public Key's data format and related verification procedure.
5 Self-signed Issuer Public Keys	Describes the self-signed Issuer Public Key's data format and related verification procedure.
6 Issuer Public Key Certificates	Describes the Issuer Public Key Certificate's data format and related verification procedure.
7 Hash codes for Transferred Public Keys	Describes how hash codes are computed and their data format.
8 File Naming Conventions	Describes the file naming conventions for files that contain keys, certificates, and hash codes.

Language Use

The spelling of English words in this manual follows the convention used for U.S. English as defined in *Merriam-Webster's Collegiate Dictionary*. MasterCard is incorporated in the United States and publishes in the United States. Therefore, this publication uses U.S. English spelling and grammar rules.

An exception to the above spelling rule concerns the spelling of proper nouns. In this case, we use the local English spelling.

Revisions

We will periodically issue revisions to this manual as we implement enhancements and changes, or as corrections are required.

With each revision, we include a "Summary of Changes" describing how the text changed. Revision markers (vertical lines in the right margin) indicate where the text changed.

Occasionally, we may publish revisions or additions to this manual in an *Operations Bulletin* or other bulletin. Revisions announced in another publication, such as a bulletin, are effective as of the date indicated in that publication, regardless of when the changes are published in this manual.

Related Information

The following documents and resources provide information related to the subjects discussed in this manual.

- *Public Key Infrastructure (PKI) — Overview.*
- *Public Key Infrastructure (PKI) — Policy*
- *Public Key Infrastructure (PKI) — Certification Authority Member Procedures.*
- *EMV2000, Integrated Circuit Card Specifications for Payment Systems – Book 2: Security & Key Management, version 4.0, December 31 2000.*
- *EMV2000, Integrated Circuit Card Specifications for Payment Systems – Book 4: Cardholder, Attendant and Acquirer Interface Requirements, version 4.0, December 31 2000.*
- *R.L. Rivest, A. Shamir and L. Adleman, A method for obtaining digital signatures and public key cryptosystems, Communications of the ACM, vol. 21, pp. 120-126 (1976).*

To order MasterCard manuals, please use the Ordering Publications service on MasterCard OnLine®, or contact the Manuals and Publications team.

Manuals and Publications
Chaussée de Tervuren, 198A
B-1410 Waterloo
Belgium

Fax: + 32 2 352 5353
e-mail: doc@mastercard.com

Support

Please address your questions to the Key Management Services Customer Support Team or the Customer Operations Services team as follows:

Key Management Services Customer Support Team

Hours of Business: 07:00-19:00 GMT/BST (UTC + 1) Monday to Friday

Phone: +44 (0) 1925 882050

Fax: +44 (0) 1925 882051

E-mail: emv_support@mastercard.com

Address: MasterCard Global Key Centre
St. Andrews House, The Links
Kelvin Close, Birchwood
Warrington, Cheshire WA3 7PB
UK

Customer Operations Services Team

Phone: 1-800-999-0363 or 1-636-722-6176
1-636-722-6292 (Spanish language support)
+32 2 352 5304 (Europe Region)

Fax: 1-636-722-7192
+32 2 352 5949 (Europe Region)

E-mail:

Canada, Caribbean, and U.S.	member_support@mastercard.com
Asia/Pacific	apms@mastercard.com
Europe	css@mastercard.com
South Asia/Middle East/Africa	emeaap@mastercard.com
Latin America (Spanish language support)	lagroup@mastercard.com

Address: MasterCard International Incorporated
Customer Operations Services
2200 MasterCard Boulevard
O'Fallon MO 63368-7263
USA

Europe
MasterCard Europe
Chaussée de Tervuren, 198A
B-1410 Waterloo
Belgium

Telex: 434800 *answerback*: 434800 ITAC UI

Member Relations Representative

Member Relations representatives assist U.S. members with marketing inquiries. They interpret member requests and requirements, analyze them, and if approved, monitor their progress through the various MasterCard departments. This does not cover support for day-to-day operational problems, which the Customer Operations Services team addresses.

To find out who your U.S. Member Relations representative is, contact your local Member Relations office:

Atlanta	1-678-459-9000
Chicago	1-847-375-4000
Purchase	1-914-249-2000
San Francisco	1-925-866-7700

For members in the Europe Region, please contact your Regional Manager.

Regional Representative

The regional representatives work out of the regional offices. Their role is to serve as intermediaries between the members and other departments in MasterCard. Members can inquire and receive responses in their own language and during their office's hours of operation.

To find out the location of the regional office serving your area, call the Customer Operations Services team at:

Phone: 1-800-999-0363 or 1-636-722-6176

1-636-722-6292 (Spanish language support)

For members in the Europe Region, please contact your Regional Manager.

Abbreviations and Terminology

The following abbreviations and terminology are used throughout this manual.

Abbreviation	Meaning
CA	Certification Authority
PK	Public Key
PKI	Public Key Infrastructure
PS	Payment System
RID	Registered Application Provider Identifier
SK	Private Key

For definitions of other key terms used in this manual, please refer to the *MasterCard Dictionary* on the Member Publications home page (on MasterCard OnLine® and the MasterCard Electronic Library CD-ROM). You also may access the MasterCard Dictionary from the main menu and bookmark pane of most manuals.

1 Interface Specification

This chapter provides an overview of communications and data formats in the key transfer interface.

Introduction

To operate the MasterCard Public Key Certification Service, the MasterCard Certification Authority and members (issuers and acquirers) need to exchange the following information:

- 'public keys'
- 'certificates' associated to 'public keys'
- 'hash codes'.

This chapter divides the key transfer interface specification into the following sections:

- Communications, explaining how the communications channel works
- Data formats, explaining how to structure the data you want to transfer.

Communications

The information listed above is transmitted in the form of electronic files. The MasterCard Certification Authority and members transfer those files using one of the following methods:

- e-mail, the fastest and preferred support
- floppy disks, delivered by courier as an alternative if e-mail is not operational.

Member requests are generated using the Member Request Tool and are sent to MasterCard as e-mail attachments. As they are protected by digital signatures, any modification to these files will be detected. Upon receipt of a member request, MasterCard automatically sends a confirmation message to the originator and proceeds with the request as soon as possible.

MasterCard responses are received as e-mail attachments. These digitally signed files are fed into the Member Request Tool for verification.

The *Certification Authority Member Procedures* manual defines the operational procedures to follow in specific circumstances.

When transferring public key data, the MasterCard Certification Authority and members must ensure dual control and data integrity in their procedures.

Member Request Tool Interface

Issuer Public Key files must be processed by the Member Request Tool to produce certification requests. Similarly, the Member Request Tool processes messages received from MasterCard, producing Payment System Public Key files or certificate files. This manual only covers the format of these files.

The interface with the Member Request Tool consists of files and hash values.

Each file consists of a string of bytes. The remainder of this manual specifies the format of each file. Chapter 8, "File Naming Conventions" specifies the file naming conventions for these files.

The MasterCard Certification Authority and members must compute hash codes using the SHA-1 algorithm, as specified in Annex B3.1 of the *EMV2000: Integrated Circuit Card Specification for Payment Systems – Book 2: Security and Key Management*. Chapter 7, "Hash Codes for Transferred Public Keys" provides further details on this subject.

Data Formats

The different file types exchanged between the member Security Officers and the MasterCard Public Key Certification Service are described in the following sections. For each file type, the default file name extension is provided.

Issuer File

Issuers must prepare their Issuer Public Keys to be submitted to the MasterCard Public Key Certification Service via the Member Request Tool in the file format shown in Table 1.1.

Table 1.1—Issuer File

File Type	Extension	Contains …
Self-signed Issuer Public Key	.sip	an Issuer Public Key concatenated with some data, including the public key validity date, the key index, and the BIN signed by the corresponding private key.

This file format is described in this manual.

Support of Issuer Public Key hash files ("*.hip") is no longer required for interfacing with the MasterCard Public Key Certification Service. Dual control over the origin and integrity of the public key certification requests is achieved by two Security Officers each sending a certificate request file.

Certification Authority Files

The MasterCard Public Key Certification Service generates files than can be stored via the Member Request Tool in the formats shown in Table 1.2.

Table 1.2—Certification Authority Files

File Type	Extension	Contains …
Issuer Public Key Certificate	.c*??* [1]	an Issuer Public Key concatenated with some data, including the certificate validity date, the key index, and the BIN signed by the specified Payment System Private Key.
Self-signed Payment System key	.sep	a Payment System Public Key concatenated with some data, including the public key validity date, the key index, and the RID signed by the corresponding private key.
Self-signed Payment System key hash	.hep	contains the hash corresponding to a self-signed Payment System key file.

[1] *??* denotes the index of the Payment System key used to generate the certificate.

These file formats are described in this manual.

Exchange Files

Although members use issuer files and Certification Authority files as described previously, the communication between the Member Request Tool and the Certification Authority is based on the exchange of authenticated files. These files are secured by the presence of electronic signatures. They are generated by the Member Request Tool using the Signature Card, or by the MasterCard Public Key Certification Service. These files are transmitted as e-mail attachments.

The files exchanged are of the types shown in Table 1.3.

Table 1.3—Exchange Files

File Type	Extension	Contains …
Certificate request	.crq	all information needed by the Certification Authority to generate an issuer certificate. It is authenticated by a signature generated by the Signature Card of the member Security Officer who issues the request.
Certificate response	.crs	a certificate for an Issuer Public Key signed by a Payment System key of the Certification Authority.
Payment System Key distribution	.pdk	Payment System Public Keys with different key indices corresponding to a given RID. The file is dedicated to a specific member Security Officer who is the only one able to decode it, using their Signature Card.

This manual does not cover the format of these files.

Format Definition

Table 1.4 details the symbols used to indicate individual data element formats.

Table 1.4—Symbols used to indicate data element formats

Symbol	Description	Example
A	Alphabetical ASCII characters ('41' Hex - '5A' Hex)	'BC' in A is coded: 0x42 0x43
A/N	Alphanumeric ASCII characters ('41' Hex - '5A' Hex) or ('30' Hex - '39')	'B8' in A/N is coded: 0x42 0x38
b	Binary 8 bit character ('00' Hex - 'FF' Hex)	'BC' in b is coded: 0xBC
cn	Compressed Numeric format (meaning BCD coded numeric digits, right+padded with F's if needed).	'123' in cn6 is coded: 0x12 0x3F 0xFF
n	Numeric format (meaning the numeric value coded in BCD).	'123' in n4 is coded: 0x01 0x23
N	Numeric ASCII characters ('30' Hex - '39' Hex)	'123' in N3 is coded: 0x31 0x32 0x33

Table 1.5—Symbols used to represent data field lengths

Symbol	Description
k	Length of the certificate subject ID
N_s	Length of the subject public key modulus
N_c	Length of the subject public key certificate
N_i	Length of the Issuer Public Key modulus
N_{ca}	Length of the Certification Authority Public Key modulus

2 High-level Data Formats

This chapter describes the main data formats transferred using the key transfer interface.

Introduction

This chapter provides a high-level description of the four predefined data formats identified in chapter 1, "Interface Specification", namely:

- MasterCard Payment System Public Keys
- Issuer Public Keys
- Issuer Public Key Certificates
- Hash codes for transferred public keys.

The MasterCard Payment System Public Keys, Issuer Public Keys and Issuer Public Key Certificates all correspond to the EMV certificate data structure.

The first three items have an EMV certificate data structure. In the first two, the EMV data structure is a 'self-signed' certificate.

Self-signed certificates provide the recipient with the following benefits:

- a measure of protection against accidental errors in transmission
- a guarantee that the certificate's originator knows the private key that corresponds to the self-signed certificate's public key.

Unlike a 'regular' certificate, a self-signed certificate contains no guarantees regarding its true origin. The Member Request Tool provides this guarantee by adding an electronic signature to the transferred file. This resulting file may be securely transferred from the Member Request Tool to the Certification Authority. It is automatically generated by the Member Request Tool and its format is not in the scope of this manual.

MasterCard Payment System Public Keys

The Certification Authority:

- signs the MasterCard Payment System Public Key using the relevant MasterCard Payment System Private Key.
- transfers MasterCard Payment System Public Keys to members via the Member Request Tool using a 'self-signed' certificate structure.

Members need to verify this self-signed certificate (and recover its contents). To do this, you need prior knowledge of the MasterCard Payment System Public Key. For this reason, the Certification Authority sends the MasterCard Payment System Public Key with the self-signed public key certificate.

The MasterCard Payment System Public Key is an EMV certificate prefixed by the MasterCard Payment System Public Key 'in clear'.

Note **The remainder of this manual refers to the 'MasterCard Payment System' Public and Private Keys simply as the 'MasterCard' Public and Private Keys.**

Issuer Public Keys

Issuers must transfer Issuer Public Keys via the Member Request Tool to the Certification Authority for certification. The issuer must use a certificate containing the self-signed Issuer Public Key. To verify the signature and recover its contents, you need prior knowledge of the Issuer Public Key. As a result, the issuer must send the Issuer Public Key within the self-signed Issuer Public Key Certificate.

The Issuer Public Key consists of an EMV certificate prefixed by the Issuer Public Key 'in clear'.

Issuer Public Key Certificates

The Certification Authority generates the Issuer Public Key Certificate. The Issuer Public Key Certificate is an EMV certificate.

Security for Transferred Elements

The Member Request Tool prevents substitution of transferred files and allows for origin authentication via an electronic signature added to the files.

Public Keys

Issuers must sign the Issuer Public Key hash codes when sending an Issuer Public Key to MasterCard. This is done using the Member Request Tool. MasterCard must verify the hash code by re-computing it using the received Issuer Public Key, and comparing the newly-computed hash code with the received signed hash code.

MasterCard encrypts the MasterCard Payment System Public Keys when sending them to members. The recipient of a Payment System Public Key must decrypt it and validate it, using the Member Request Tool.

Certificates

The integrity of the received certificate must be checked by using the Payment System Public Key to verify its signature.

3 Certificates and Associated Data

This chapter provides an overview of how certificates are constructed and verified, and a detailed description of their content.

Introduction

All certificates that the Certification Authority and members transfer must conform to the 'EMV' format (as specified in Book 2 of the *EMV2000 Integrated Circuit Card Specification for Payment Systems*). This chapter explains the process of generating an EMV certificate.

Book 2 of the EMV2000 specifications gives separate descriptions for the process of generating the following two items:

- Issuer Public Key Certificate
- ICC Public Key Certificate.

When the Certification Authority or an issuer constructs a certificate, several of the certificate's data fields are specific to the certificate type. Issuer Public Key Certificates and ICC Public Key Certificates both contain the same overall structure and corresponding sets of data fields. However, an Issuer Public Key Certificate uses an Issuer Public Key Length data field, whereas an ICC Public Key Certificate uses its own specific ICC Public Key Length data field.

To explain the structure of EMV certificates, this chapter focuses on areas common to both Issuer Public Key Certificates and ICC Public Key Certificates. When you assign the appropriate specific values to the two certificates, both will follow the same generic structure on a bit-for-bit basis.

In this chapter, we use the word 'subject' to identify sections of EMV certificates that contain content specific to the individual certificate type. For example, 'subject' corresponds to 'Issuer' in an Issuer Public Key Certificate, and to 'ICC' in an ICC Public Key Certificate.

Constructing Certificates

The Certification Authority and issuers construct EMV certificates by assembling the sequence of data fields for input to the hash function (as specified in Book 2 of the EMV2000 specifications, Tables 1 and 7). In this chapter, Table 3.1 provides a summary of this information.

Apply the signature scheme (as specified in Annex A2.1 of Book 2 of the EMV2000 specifications) to the sequence of data fields specified in Table 3.1 using the private key of the entity that is signing the certificate.

More specifically, the entity signing the certificate must perform the following steps:

1. Input the complete set of data fields to the hash function in use (which must be SHA-1 – see Annex B3.1 of Book 2 of the EMV2000 specifications) and obtain a 20-byte Hash Result.
2. Concatenate all but the last two data fields with the 20-byte Hash Result (just computed) and prefix and post-fix the Hash Result with fixed values before inputting the mathematical signature function.
3. The signature function outputs a string of N_c bytes. This string constitutes a data field known as the 'Subject Public Key Certificate' (see Tables 3 and 8 of Book 2 of the EMV2000 specifications).

Note

Due to the Subject Public Key's length, the signature algorithm cannot recover all of the Subject Public Key by itself. The signature algorithm specified in Annex B2.1 of Book 2 of the EMV2000 specifications only allows the recovery of data strings with lengths of up to N_C -22 bytes.

To recover the Subject Public Key, the process divides the data string into two parts. By concatenating all but the last two of the data fields listed in Table 3.1, the algorithm outputs a string of precisely N_C -22 bytes. This recovers as much of the Subject Public Key as is possible from the data to be signed.

To enable recipients to verify the Subject Public Key Certificates, the sender must always transmit both self-signed and signed 'Subject' Public Key Certificates with the last two data fields listed in Table 3.1. (For more information on this, see Book 2 of the EMV2000 specifications, Tables 3 and 8). The sender must also send the following data:

- The identity of the signer of the certificate
- The identifier of the key used to sign the certificate

- (Only if the certificate is self-signed) further information, in particular the third to last data field from Table 3.1. (See chapter 4, "Self-signed Payment System Public Keys", chapter 5, "Self-signed Issuer Public Keys" and chapter 6, "Issuer Public Key Certificates" for further details of precisely what information to send with a certificate, including the exact order of the data elements.)

Table 3.1—Data Fields Used to Create a Certificate

Field Name	Length	Description	Format
Certificate Format	1	Dependent on certificate type (see Table 3.2)	b
ID of Certificate Subject	Dependent on certificate type (see Table 3.2) - denoted by k	Dependent on certificate type (see Table 3.2)	Dependent on certificate type (see Table 3.2)
Certificate Expiry Date	2	MMYY after which certificate is invalid	n4
Certificate Serial Number	3	Dependent on certificate type (see Table 3.2)	b
Hash Algorithm Indicator	1	Indicates the hash algorithm used to create the certificate (see EMV Specs)	b
Subject Public Key Algorithm Indicator	1	Indicates the algorithm to be used with the Subject Public Key (see EMV Specs)	b
Subject Public Key Length	1	The length of the Subject Public Key Modulus in bytes (N_s)	b
Subject Public Key Exponent Length	1	The length of the Subject Public Key Exponent in bytes	b
Subject Public Key or Leftmost Digits of the Subject Public Key	N_c - 32 - k	If $N_s \leq N_c$ -32 - k then this field contains the complete modulus for the Subject Public Key, right-padded with N_c -32 - k - N_s bytes of value 'BB' (hexadecimal) If $N_s > N_c$ - 32 - k then this field contains the N_c - 32 - k most significant bytes of the modulus for the Subject Public Key	b
Subject Public Key Remainder	0, or N_s - N_c + 32 +k	This field is only present if $N_s > N_c$ - 32 - k, and if present this field contains the N_s - N_c + 32 + k least significant bytes of the modulus for the Subject Public Key	b
Subject Public Key Exponent	1 or 3	Value 3 or 2^{16}+1	b

Verification of Certificates

To verify a certificate, the recipient must perform the following steps:

1. Submit the following data to the verification process described in Annex B2.1.3 of Book 2 of the EMV2000 Specifications:
   - the 'Subject Public Key Certificate' (a string of N_C bytes)
   - the last two data fields in Table 3.1, namely the 'Subject Public Key Remainder' and the 'Subject Public Key Exponent'
   - the signer of the certificate's public key (namely the signer's Public Key Modulus and the signer's Public Key Exponent)
2. The verification process will yield one of two results: success or failure. If the verification process is successful it will yield verified versions of all the data fields in Table 3.1.

Chapter 4, "Self-signed Payment System Public Keys", chapter 5, "Self-signed Issuer Public Keys" and chapter 6, "Issuer Public Key Certificates" give detailed guidance on the verification process for each of the three types of certificate.

Values for Specific Certificate Types

Certain data fields used to construct certificates vary depending on the certificate's type. This section explains three certificate types:

- self-signed MasterCard Payment System Public Keys
- self-signed Issuer Public Keys
- Issuer Public Keys certified by MasterCard.

Table 3.2 indicates the values used for each of these three types of certificate.

Table 3.2—Certificate Data Field Values

Certificate Type	Self-signed MasterCard Public Keys	Self-signed Issuer Public Keys	Issuer Public Keys Certified by MasterCard
Certificate Format (1 byte)	10 (hex)	11 (hex)	02 (hex)
Certificate Subject ID length (k)	5	4	4
Certificate Subject ID description	The 'Registered Application Provider Identifier' (RID) for MasterCard	Leftmost 3-8 digits of the PAN right-padded with hexadecimal 'F' characters	Leftmost 3-8 digits of the PAN right-padded with hexadecimal 'F' characters
Certificate Subject ID Format	b	cn8	cn8
Certificate Serial Number Description	3-byte value chosen by MasterCard	3-byte value chosen by the issuer	3-byte value assigned by the Certification Authority to uniquely identify this certificate

The creator of the certificate always chooses the 3-byte value to insert in the Certificate Serial Number field. Thus MasterCard selects the Certificate Serial Number to insert in self-signed certificates for MasterCard Public Keys. Likewise, issuers select the Certificate Serial Number to insert in self-signed certificates for Issuer Public Keys.

4 Self-signed Payment System Public Keys

This chapter describes the self-signed Payment System Public Key's data format and related verification procedure.

Introduction

This chapter specifies the precise data format used when MasterCard transfers a Payment System Public Key to an issuer or acquirer. It also describes the procedure that the recipient of the Payment System Public Key must follow.

Construction of Self-signed Payment System Public Keys

As explained in chapter 2, "High-level Data Formats", the Member Request Tool can store the MasterCard Public Key in two parts:

- a (self-signed) EMV certificate
- the two sections of the MasterCard Public Key 'in clear'.

To make it more convenient for the recipients to process this data, additional identifying information also prefixes the MasterCard Public Key.

Chapter 3, "Certificates and Associated Data" describes how self-signed EMV certificates are constructed.

MasterCard signs the MasterCard Public Key using the MasterCard Private Key. Hence in this case $N_C = N_S = N_{CA}$ and k = 5, (where N_{CA} is the length (in bytes) of the MasterCard Public Key Modulus). The resulting data field, known as the MasterCard Public Key Certificate, consists of a string of N_{CA} bytes.

MasterCard transfers all MasterCard Public Key Certificates with the following two additional data fields:

- the MasterCard Public Key Remainder (a string of 37 bytes)
- the MasterCard Public Key Exponent.

MasterCard also sends MasterCard Public Key information items 'in clear' with the certificate. The information items are the following fields specified in Table 3-1:

- ID of Certificate Subject (to identify MasterCard Public Key)
- MasterCard Public Key Index (to help identify the key)
- MasterCard Public Key Algorithm Indicator (to identify which algorithm the key should be used with)
- MasterCard Public Key Length (to support verification)
- MasterCard Public Key Exponent Length (to support verification)
- Leftmost digits of the MasterCard Public Key (to support verification).

Table 4.1 lists the complete set of nine data fields that constitute the transfer format for MasterCard Public Keys. The data fields help the recipient process the MasterCard Public Key data as follows:

- The first field distinguishes the key as a MasterCard Public Key.
- The second field identifies the particular MasterCard Public Key.
- The third field identifies the signature algorithm to use with the key. (This field's content is currently fixed, but the field is included here as it may be used more flexibly in the future.)
- The next three fields provide public key information 'in clear'.
- The last three fields provide the certificate information.

Table 4.1—Transfer Format: MasterCard Public Key (Self-signed) and Associated Data

Field Name	Length	Description	Format
ID of Certificate Subject	5	The 'Registered Application Provider Identifier' (RID) for MasterCard	b
MasterCard Public Key Index	1	The MasterCard Public Key Index uniquely identifies a MasterCard Public Key	b
MasterCard Public Key Algorithm Indicator	1	Indicates the algorithm to be used with the MasterCard Public Key, set to '01' (hex) (see EMV specs)	b
MasterCard Public Key Length	1	Length of the MasterCard Public Key Modulus (equal to N_{CA})	b
MasterCard Public Key Exponent Length	1	Length of the MasterCard Public Key Exponent (equal to 1)	b
Leftmost Digits of MasterCard Public Key	N_{CA} – 37	N_{CA} - 37 most significant bytes of the MasterCard Public Key Modulus	b
MasterCard Public Key Remainder	37	37 least significant bytes of the MasterCard Public Key Modulus	b
MasterCard Public Key Exponent	1	MasterCard Public Key Exponent – value 3	b
MasterCard Public Key Certificate	N_{CA}	Output of signature algorithm	b

Verification of Self-signed MasterCard Public Keys

To verify a MasterCard Public Key, members must perform the following steps. These steps ensure that you recover the public key as well as verifying the received public key:

1. Process the ID of Certificate Subject (first of the nine fields in Table 4.1) to recover the RID. You must check this to verify that it corresponds to the expected RID for MasterCard. If it differs from the expected RID for MasterCard, you must reject the public key.
2. Compare the MasterCard Public Key Index (second of the nine fields in Table 4.1) with the MasterCard Public Key Indexes for any MasterCard Public Keys already stored by the issuer. If there is a clash, you must reject the public key.
3. Check the MasterCard Public Key Algorithm Indicator (third of the nine fields in Table 4.1) to verify that it is equal to '01'. If it is not equal to '01', you must reject the public key. (Should alternative public key algorithms be permitted at some point in the future, this check will be changed).
4. Check the MasterCard Public Key Length (the fourth of the nine fields listed in Table 4.1) to verify that it is within the acceptable values for a MasterCard Public Key Modulus length – N_{CA} denotes this length value. If this check fails, you must reject the public key.
5. Check the entire self-signed data structure to ensure that it contains exactly $2N_{CA}+10$ bytes – if it does not, you must reject the public key.
6. Check the MasterCard Public Key Exponent Length (the fifth field from Table 4.1) to verify that it equals '01'. If this check fails, you must reject the public key.
7. Process the next N_{CA} bytes of transferred data (the sixth and seventh fields from Table 4.1) to yield the MasterCard Public Key Modulus.
8. Process the following byte (the eighth field from Table 4.1) to yield the MasterCard Public Key Exponent. You must check that this value is within the set of acceptable values for a Payment System Public Key Exponent – if it is not, you must reject the public key.
9. Perform the recovery function specified in Annex A2.1 of Book 2 of the EMV2000 specifications on the final string of N_{CA} bytes. To do this, use the MasterCard Public Key that you recovered in the previous two steps, in conjunction with the cryptographic function specified in Annex B2.1.2.3 of the EMV2000 specifications. You will recover N_{CA} bytes of data according to the format given in Table 4.2.

10. Check the Recovered Data Header and Trailer. If one of the following checks fails, you must reject the public key:
    - Header is equal to '6A' (hex)
    - Trailer is equal to 'BC' (hex)
11. Check the Certificate Format field. If it is not equal to '10' (hex) (see also Table 3-2), you must reject the public key.
12. Concatenate the following eleven data elements, from left to right: the second to the tenth data elements listed in Table 4.2 ('Certificate Format' through to 'Leftmost Digits of the MasterCard Public Key'), followed by the MasterCard Public Key Remainder and the Subject Public Key Exponent (as originally transferred).
13. Apply the hash algorithm (indicated by the Hash Algorithm Indicator) to the output of the previous step to produce the Hash Result.
14. Compare the calculated Hash Result (from the previous step) with the recovered Hash Result. If they are not the same, you must reject the public key.
15. Check the certificate's ID of Certificate Subject (the third element in Table 4.2) to ensure that it matches the value sent with the certificate (first field in Table 4.1). If they are not the same, you must reject the public key.
16. The last day of the month specified in the Certificate Expiry Date must be equal to, or later than, the current date. If it is not, the self-signed certificate has expired, and you must reject the public key.
17. This is an optional step. You may check the concatenation of the following fields: ID of Certificate Subject, MasterCard Public Key Index, and Certificate Serial Number. If you consider that this data is invalid, you must reject the public key.
18. Check the MasterCard Public Key Algorithm Indicator for the expected value, namely '01' (hex). If its value is not '01' (hex), you must reject the public key.
19. Check that the following three data elements: 'MasterCard Public Key Length', 'MasterCard Public Key Exponent Length', and 'Leftmost digits of MasterCard Public Key' (the eighth, ninth and tenth fields in Table 4.2) match the values sent with the certificate (the fourth, fifth and sixth fields in Table 4.1). If these values do not match, you must reject the public key.
20. Re-compute the hash code (the 'Certification Authority Public Key Check Sum'), as described in chapter 7, "Hash Codes for Transferred Public Keys", using the received MasterCard Public Key Index and the values recovered from the certificate.

21. The calculated hash code (as above) must match the separately transferred hash code. If the two differ, you must reject the public key.

If all of the above checks proceed correctly, you may accept the MasterCard Public Key. Issuers will probably wish to retain the data items specified in Table 4.3 as a 'full set' of 'MasterCard Public Key Information'.

Most of these checks are performed automatically by the Member Request Tool when it receives the Payment System Public Keys.

Table 4.2—Format of Data Recovered from a Self-signed MasterCard Public Key Certificate

Field Name	Length	Description	Format
Recovered Data Header	1	Hexadecimal value '6A' (see EMV specs)	b
Certificate Format	1	Equal to '10' (hex) - see also Table 3-2	b
ID of Certificate Subject	5	The MasterCard RID	b
Certificate Expiry Date	2	MMYY after which certificate is invalid	n4
Certificate Serial Number	3	A 3-byte value inserted by MasterCard	b
Hash Algorithm Indicator	1	Indicates the hash algorithm used to create the certificate (see EMV specs)	b
MasterCard Public Key Algorithm Indicator	1	Indicates the algorithm to be used with the MasterCard Public Key, equal to '01' (hex) (see EMV specs)	b
MasterCard Public Key Length	1	The length of the MasterCard Public Key Modulus in bytes (N_{CA})	b
MasterCard Public Key Exponent Length	1	The length of the MasterCard Public Key Exponent in bytes (equal to 1)	b
Leftmost Digits of the MasterCard Public Key	N_{CA} - 37	This field contains the N_{CA} – 37 most significant bytes of the modulus for the MasterCard Public Key	b
Hash Result	20	Hash of the MasterCard Public Key and its associated information	b
Recovered Data Trailer	1	Hexadecimal value 'BC' (see EMV specs)	b

Table 4.3—Retained MasterCard Public Key Information

Field Name	Length	Description	Format
ID of Certificate Subject	5	The MasterCard RID	b
MasterCard Public Key Index	1	An identifier for the public key	b
Certificate Expiry Date	2	MMYY after which certificate is invalid	n4
Certificate Serial Number	3	3-byte value chosen by MasterCard	b
MasterCard Public Key Length	1	The length of the MasterCard Public Key Modulus in bytes (N_{CA})	b
MasterCard Public Key Exponent Length	1	The length of the MasterCard Public Key Exponent in bytes (equal to 1)	b
Leftmost Digits of the MasterCard Public Key	N_{CA} -37	This field contains the N_{CA} –37 most significant bytes of the modulus for the MasterCard Public Key	b
MasterCard Public Key Remainder	37	37 least significant bytes of the MasterCard Public Key Modulus	b
MasterCard Public Key exponent	1	MasterCard Public Key Exponent – value 3	b

5 Self-signed Issuer Public Keys

This chapter describes the self-signed Issuer Public Key's data format and related verification procedure.

Introduction

This chapter specifies the precise data format to be used when an issuer transfers an Issuer Public Key to MasterCard. It also describes the procedure that the recipient of the Issuer Public Key must follow.

Construction of Self-signed Issuer Public Keys

As explained in chapter 2, "High-level Data Formats", an issuer must import the Issuer Public Key into the Member Request Tool in two parts:

- a (self-signed) EMV certificate
- the Issuer Public Key 'in clear'.

Chapter 3, "Certificates and Associated Data" describes how self-signed EMV certificates are constructed. In this chapter, the 'Subject Public Key' is the 'Issuer Public Key'.

This data must be prefixed with certain identifying information. Specifically, the issuer must choose a unique three byte 'Issuer Public Key File Index' for each of the following file pairs they generate:

- public key file (extension '.sip')
- hash code file[1] (extension '.hip').

Issuers must ensure that the same Issuer Public Key File Index is never used twice.

The issuer uses the "Issuer Public Key File Index" to identify both files and Issuer Public Keys in communications with MasterCard.

The issuer must construct the self-signed EMV certificate as described in chapter 3, "Certificates and Associated Data", with the 'Subject Public Key' being the 'Issuer Public Key'. The issuer must sign the self-signed EMV certificate using the Issuer Private Key. Hence in this case $N_C = N_S = N_I$, and k = 4, (where N_I is the length – in bytes – of the Issuer Public Key Modulus). The resulting data field, known as the Issuer Public Key Certificate, consists of a string of N_I bytes.

[1] The '.hip' file format is only described for compatibility with some existing hardware requiring the presentation of the two files. The Member Request Tool does not require the '.hip' file although the value of the hash should be known to the member Security Officer.

The issuer must transfer all Issuer Public Key Certificates with the following two data fields from Table 3-1:

- Issuer Public Key Remainder (a string of 36 bytes)
- Issuer Public Key Exponent.

The issuer must send 'Issuer Public Key information' to MasterCard 'in clear' with the certificate. This 'Issuer Public Key information' must include the following fields (all but the Issuer Public Key Index are listed in Table 3-1):

- ID of Certificate Subject (to identify the key)
- Issuer Public Key Index (to assist in key identification)
- Issuer Public Key Algorithm Indicator (to identify which algorithm the key should be used with)
- Issuer Public Key Length (to support verification)
- Issuer Public Key Exponent Length (to support verification)
- Leftmost Digits of the Issuer Public Key (to support verification).

Table 5.1 lists the complete set of nine data fields that constitute the transfer format for Issuer Public Keys. The data fields help the recipient process the Issuer Public Key data as follows:

- The first two fields distinguish the key from other keys that the issuer supplies.
- The third field identifies the signature algorithm with which this key must be used. (This is currently fixed, but the field is included here for possible future use.)
- The next three fields provide public key information 'in clear'.
- The last three fields provide the certificate information.

Note **The values of the 'ID of Certificate Subject' and 'Certificate Expiry Date' fields of the self-signed Issuer Public Key affect the values that get stored in the corresponding fields of the resulting Issuer Public Key Certificate. The Certificate Authority also signs the Issuer Public Key Certificate. For more information, see chapter 6, "Issuer Public Key Certificates".**

The issuer must perform the following actions relating to the 'ID of Certificate Subject' and 'Certificate Expiry Date' fields:

- Set the 'ID of Certificate Subject' field's value to an appropriate set of 3-8 PAN digits. The Certification Authority includes this value in the 'ID of Certificate Subject' field of the resulting Issuer Public Key Certificate.
- Set the 'Certificate Expiry Date' field's value to equal the intended expiry date for this key. The Certification Authority includes this value in the 'Certificate Expiry Date' field of the resulting Issuer Public Key Certificate (unless the date is later than the expiry date of the Payment System key set by MasterCard.)

Table 5.1 lists the fields of the Issuer Public Key with their descriptions. (The fields correspond precisely to those labeled in Book 2 of the EMV2000 specifications.)

Table 5.1—Transfer Format: Issuer Public Key (Self-signed) and Associated Data

Field Name	Length	Description	Format
ID of Certificate Subject	4	Leftmost 3-8 digits from the PAN, right-padded with hexadecimal 'F' characters	cn8
Issuer Public Key File Index	3	Number, chosen by the issuer, which uniquely identifies the public key file	b
Issuer Public Key Algorithm Indicator	1	Indicates the algorithm to be used with the Issuer Public Key, set to '01' (hex)	b
Issuer Public Key Length	1	Length of the Issuer Public Key Modulus (equal to N_I)	b
Issuer Public Key Exponent Length	1	Length of the Issuer Public Key Exponent (equal to 1 or 3)	b
Leftmost Digits of Issuer Public Key	N_I – 36	N_I -36 most significant bytes of the Issuer Public Key Modulus	b
Issuer Public Key Remainder	36	36 least significant bytes of the Issuer Public Key Modulus	b
Issuer Public Key Exponent	1 or 3	Issuer Public Key Exponent – value 3 or $2^{16}+1$	b
Issuer Public Key Certificate	N_I	Output of signature algorithm	b

Verification of Self-signed Issuer Public Keys

To verify an Issuer Public Key, the Certification Authority must perform the following steps. These steps ensure that we verify the received public key as well as recovering it. You must ensure that you format your public key files according to the specifications so that they will pass our verification procedures.

1. Process the ID of Certificate Subject (first of the nine fields in Table 5.1) to recover the first 3-8 digits of the issuer PAN. We check the PAN to verify that it corresponds to a permissible value. If the PAN does not correspond to a permissible value, we reject the public key.
2. Check the Subject Public Key Algorithm Indicator (third of the nine fields in Table 5.1) to verify that its value is '01' (hex). If the value is not '01', we reject the public key. (Should alternative algorithms be permitted at some point in the future, this check will be changed.)
3. Check the Subject Public Key Length (fourth of the nine fields in Table 5.1) to verify that it is acceptable for an Issuer Public Key Modulus length –N_I denotes this value. If this check fails, we reject the public key.
4. Check the entire self-signed data structure to ensure that it contains $2N_I$ + 11 or $2N_I$ + 13 bytes – if it does not, we reject the public key.
5. Check the Subject Public Key Exponent Length (the fifth field from Table 5.1) to verify that it is within the acceptable values for an Issuer Public Key Exponent. If this check fails, we reject the public key.
6. Process the next N_I bytes of transferred data (the sixth and seventh fields from Table 5.1) to yield the Issuer Public Key Modulus.
7. Process the following string of 1 or 3 bytes (the eighth field from Table 5.1, and of length specified by the fifth field of Table 5.1) to yield the Issuer Public Key Exponent. We check this value to see whether it is within the set of acceptable values for an Issuer Public Key Exponent – if it is not, we reject the public key.
8. Perform the recovery function specified in Annex B2.1 of Book 2 of the EMV2000 specifications on the final string of N_I bytes. To do this, we use the Issuer Public Key recovered in the previous two steps, in conjunction with the cryptographic function specified in Annex B2.1 of the Book 2 of the EMV2000 specifications. We recover N_I bytes of data according to the format given in Table 5.2.
9. Check the Recovered Data Header and Trailer. If one of the following checks fails, we reject the public key:
   - Header is equal to '6A' (hex)
   - Trailer is equal to 'BC' (hex)

10. Check the Certificate Format field. If it is not '11' (hex), (see Table 3-2), we reject the public key.
11. Concatenate the following eleven data elements, from left to right: the second to the tenth data elements in Table 5.2 ('Certificate Format' through to 'Leftmost Digits of the Issuer Public Key'), followed by the Issuer Public Key Remainder and the Issuer Public Key Exponent (as originally transferred).
12. Apply the hash algorithm (indicated by the Hash Algorithm Indicator) to the output of the previous step to produce the Hash Result.
13. Compare the calculated hash result (from the previous step) with the recovered Hash Result. If they are not the same, we reject the public key.
14. Check the ID of Certificate Subject recovered from the certificate (third element in Table 5.2) to ensure that it matches the value sent with the certificate (first field in Table 5.1). If they are not the same, we reject the public key.
15. The last day of the month specified in the Certificate Expiry Date must be equal to, or later than, the current date. If not, the self-signed certificate has expired and we reject the public key.
16. This is an optional step. We may check the concatenation of the following fields: ID of Certificate Subject, Issuer Public Key Index, and Certificate Serial Number. If we consider this data invalid, we reject the public key.
17. Check the Issuer Public Key Algorithm Indicator for the expected value, i.e. '01' (hex). If its value is not '01' (hex), we reject the public key.
18. Check that the following three data elements: 'Subject Public Key Length', 'Subject Public Key Exponent Length', and 'Leftmost digits of Subject Public Key' (eight, ninth and tenth fields in Table 5.2) match the values sent with the certificate (fourth, fifth and sixth fields in Table 5.1). If they are not the same, we reject the public key.
19. Re-compute the hash code (or 'checksum') as described in chapter 7, "Hash Codes for Transferred Public Keys", using the received Issuer Public Key Index, and the values recovered from the certificate.
20. The calculated hash code (as above) must match the hash code that the issuer transferred separately. If the two differ, we reject the public key.

If all the above checks proceed correctly then we may accept the Issuer Public Key. MasterCard may, at least temporarily, retain the data items specified in Table 5.3 as a 'full set' of Issuer Public Key Information.

Most of these checks are automatically performed by the Member Request Tool when it imports the issuer public key.

Table 5.2—Format of Data Recovered from the Self-signed Issuer Public Key Certificate

Field Name	Length	Description	Format
Recovered Data Header	1	Hexadecimal value '6A' (see EMV specs)	b
Certificate Format	1	11 (hex) – see also Table 3-2	b
ID of Certificate Subject	4	Leftmost 3-8 digits from the PAN, right-padded with hexadecimal 'F' characters	b
Certificate Expiry Date	2	MMYY after which certificate is invalid	cn8
Certificate Serial Number	3	A 3-byte value chosen by the issuer	b
Hash Algorithm Indicator	1	Indicates the hash algorithm used to create the certificate (see EMV specs)	b
Subject Public Key Algorithm Indicator	1	Indicates the algorithm to be used with the Issuer Public Key ('01') (see EMV specs)	b
Subject Public Key Length	1	The length of the Issuer Public Key Modulus in bytes (N_I)	b
Subject Public Key Exponent Length	1	The length of the Issuer Public Key Exponent in bytes	b
Leftmost Digits of the Subject Public Key	N_I - 36	This field contains the N_I - 36 most significant bytes of the modulus for the Issuer Public Key	b
Hash Result	20	Hash of the Issuer Public Key and its associated information	b
Recovered Data Trailer	1	Hexadecimal value 'BC' (see EMV specs)	b

Table 5.3—Retained Issuer Public Key Information

Field Name	Length	Description	Format
ID of Certificate Subject	4	Leftmost 3-8 digits from the PAN, right-padded with hexadecimal 'F' characters	cn8
Issuer Public Key File Index	3	Number, chosen by the issuer, which uniquely identifies the public key file	b
Certificate Expiry Date	2	MMYY after which certificate is invalid	n4
Subject Public Key Length	1	The length of the Issuer Public Key Modulus in bytes (N_I)	b
Subject Public Key Exponent Length	1	The length of the Issuer Public Key Exponent in bytes	b
Leftmost Digits of the Subject Public Key	N_I -36	This field contains the N_I – 36 most significant bytes of the modulus for the Issuer Public key	b
Subject Public Key Remainder	36	36 least significant bytes of the Issuer Public Key Modulus	b
Subject Public Key Exponent	1 or 3	Issuer Public Key Exponent – value 3 or $2^{16}+1$	b

6 Issuer Public Key Certificates

This chapter describes the Issuer Public Key Certificate's data format and related verification procedure.

Introduction

This chapter specifies the precise data format used when MasterCard transfers an Issuer Public Key Certificate to an issuer. It also describes the verification procedure to follow at the institution receiving the Issuer Public Key Certificate must follow.

Note **Book 2 of the *EMV2000 specifications* describes the verification procedure for an Issuer Public Key Certificate. As a result, this chapter only describes that part of the issuer verification process not already described in the *EMV2000 specifications*.**

Construction of Issuer Public Key Certificates

Chapter 2, "High-level Data Formats" explains that issuers must import the Issuer Public Key into the Member Request Tool in the form of an EMV certificate. They must also prefix the Issuer Public Key with additional identifying information.

The Member Request Tool exports an EMV certificate constructed as described in chapter 3, "Certificates and Associated Data". The 'Subject' Public Key of the EMV Certificate is the Issuer Public Key. The EMV Certificate is signed using the MasterCard Private Key. Hence in this case $N_C = N_{CA}$, $N_S = N_I$ and k = 4, (where N_{CA} is the length (in bytes) of the MasterCard Public Key Modulus, and N_I is the length (in bytes) of the Issuer Public Key Modulus). The resulting data field, the 'Issuer Public Key Certificate', is a string of N_{CA} bytes.

The issuer must import the Issuer Public Key Certificate with the following two data fields:

- Issuer Public Key Remainder (if present, a string of $N_I - N_{CA} + 36$ bytes)
- Issuer Public Key Exponent.

The issuer must import the following Issuer Public Key information items 'in clear' with the certificate (which correspond to the fields listed in Table 3-1):

- ID of Certificate Subject (to assist in key identification)
- Issuer Public Key Index (to assist in key identification)
- MasterCard Public Key Index (to indicate which MasterCard Private Key has been used to generate the certificate).

Note **The Issuer Public Key File Index is not an EMV Data Element (unlike the MasterCard Public Key Index). Issuers use the Issuer Public Key File Index to identify individual public key files.**

Table 6.1 lists the complete set of six data fields that constitute the transfer format for Issuer Public Key Certificates:

- The first two fields match the certificate to the key originally supplied by the issuer.
- The third field indicates which MasterCard Private Key was used to sign the certificate.
- The last three fields provide the certificate information.

Note **The values in the 'ID of Certificate Subject' and 'Certificate Expiry Date' fields of the certificate are based on values stored in the corresponding fields of the self-signed certificate for this public key supplied to the issuer. For more information, refer to chapter 5, "Self-signed Issuer Public Keys".**

The Certification Authority, in generating the Issuer Public Key Certificate, will assign the following values to the 'ID of Certificate Subject' and 'Certificate Expiry Date' fields:

- The 'ID of Certificate Subject' field will be equal to the 'ID of Certificate Subject' field in the self-signed certificate supplied by the issuer.
- The 'Certificate Expiry Date' field will be equal to the 'Certificate Expiry Date' field supplied by the issuer in the self-signed certificate. If that date is later than the expiry date of the Payment System key, the 'Certificate Expiry Date' field will be equal to the Payment System key expiry date.

Table 6.1 lists the fields of the Issuer Public Key Certificate alongside their lengths, descriptions, and formats. (The fields correspond precisely to those labeled in Book 2 of the EMV2000 specifications.)

Note **The MasterCard Public Key Index will be included since MasterCard may have more than one key pair in use at any one time. Conceivably, the same Issuer Public Key could have two or more certificates generated using distinct MasterCard Private Keys; in such a case, the MasterCard Public Key Index can be used to distinguish between the two certificates.**

Table 6.1—Transfer Format: Issuer Public Key Certificate and Associated Date

Field Name	Length	Description	Format
ID of Certificate Subject	4	Leftmost 3-8 digits from the PAN, right-padded with hexadecimal 'F' characters	cn8
Issuer Public Key File Index	3	Number, chosen by the issuer, which uniquely identifies the public key file	b
MasterCard Public Key Index	1	The MasterCard Public Key Index uniquely identifies a MasterCard Public Key	b
Issuer Public Key Remainder	0 or $N_I - N_{CA} + 36$	This field is only present if $N_I > N_{CA} - 36$, and consists of the $N_I - N_{CA} + 36$ least significant bytes of the Issuer Public Key Modulus	b
Issuer Public Key Exponent	1 or 3	Issuer Public Key Exponent – value 3 or $2^{16}+1$	b
Issuer Public Key Certificate	N_{CA}	Output of signature algorithm (the Issuer Public Key Certificate)	b

Verification of Issuer Public Key Certificates

When you (as an issuer) receive an Issuer Public Key Certificate from MasterCard, you must perform the following steps to verify the Issuer Public Key Certificate:

1. Check the Issuer Public Key Index (second of the six fields in Table 6.1) to verify that the Issuer Public Key Certificate corresponds to a public key:
   - which you have already sent to MasterCard for certification, and,
   - for which a certificate has not previously been received from MasterCard.

   If the Issuer Public Key Certificate does not meet these conditions, you must reject the certificate.
2. Process the ID of Certificate Subject field (first of the six fields in Table 6.1) to recover the leftmost 3-8 digits of the issuer PAN. Check the PAN to verify that it corresponds to the expected value for the particular Issuer Public Key Index. If it does not correspond to the expected value, you must reject the certificate.
3. Check the MasterCard Public Key Index (third of the six fields in Table 6.1) to verify that you possess a MasterCard Public Key with this Public Key Index. If you do not, you must reject the certificate.
4. Follow the procedure specified in Book 2 of the EMV2000 specifications for the verification of an Issuer Public Key Certificate and retrieval of an Issuer Public Key. You must specifically follow all of the steps in Section 5.3 of Book 2 of the EMV2000 specifications. In this process, you must use the public key identified by the MasterCard Public Index that was sent with the certificate. If any step fails, you must reject the certificate.
5. Compare all parts of the recovered Issuer Public Key (Length, Modulus and Exponent) with the corresponding values originally sent to MasterCard for certification. You must match the sent and received versions of the Issuer Public Key Index. If they do not match, you must reject the certificate.

If all the above checks proceed correctly then you may accept the Issuer Public Key Certificate.

The Member Request Tool automatically performs many of these tests when the certificate is received, providing that the Payment System public key is known.

7 Hash Codes for Transferred Public Keys

This chapter describes how hash codes are computed and their data format.

Introduction

This chapter describes how to compute hash codes. MasterCard and members use hash codes in the following situations:

- when MasterCard transfers a self-signed MasterCard Public Key to members, the Member Request Tool validates the key, then outputs its hash code.
- when an issuer generates a self-signed Issuer Public Key to be sent to MasterCard for certification, it must also generate a hash code. This hash code is used by the issuer Security Officers to authenticate the Issuer Public Key for which they generate certificate requests using the Member Request Tool.

Note

The hash codes described in this chapter are calculated using a different input date from the hash code mentioned in chapter 4, "Self-signed Payment System Public Keys" and chapter 5, "Self-signed Issuer Public Keys". The hash code described in chapter 4 is included in the self-signed Payment System Public Key Certificate, and the one described in chapter 5 is included in a self-signed Issuer Public Key Certificate.

Calculation of Hash Codes

You must compute hash codes over the fields specified in Table 7.1 and Table 7.2.

MasterCard Public Keys

Table 7.1 lists the data elements that MasterCard uses to compute the hash code for a MasterCard Public Key. MasterCard concatenates the data elements in the order specified in Table 7.1 to yield a string of N_{CA} + 7 bytes. MasterCard inputs the resulting string of bytes to the SHA-1 hash function (see Annex B3.1 of Book 2 of the EMV2000 specifications). The hash function then outputs the 20-byte hash code.

Table 7.1—Data Fields for MasterCard Public Key Hash Code Calculation

Field Name	Length	Description	Format
ID of Certificate Subject	5	The MasterCard RID	b
MasterCard Public Key Index	1	Uniquely identifies the MasterCard Public Key	b
Leftmost Digits of the MasterCard Public Key	N_{CA} – 37	N_{CA} - 37 most significant bytes of the MasterCard Public Key Modulus	b
MasterCard Public Key Remainder	37	37 least significant bytes of the MasterCard Public Key Modulus	b
MasterCard Public Key Exponent	1	MasterCard Public Key Exponent – value 3	b

For a MasterCard Public Key, the hash code is equal to the 'Certification Authority Public Key Checksum', as specified in Part 2 of Book 4 of the EMV2000 specifications.

Issuer Public Keys

Table 7.2 lists the data elements that the issuer must use to compute the hash code for an Issuer Public Key. The issuer must concatenate the data elements in the order specified in Table 7.2 to yield a string of $N_I + 8$ or $N_I + 10$ bytes. The issuer must input the resulting string of bytes to the SHA-1 hash function (see Annex B3.1 of Book 2 of the EMV2000 specifications). The hash function then outputs the hash code.

Table 7.2—Data Fields for Issuer Public Key Hash Code Calculation

Field Name	Length	Description	Format
ID of Certificate Subject	4	Leftmost 3-8 digits from the PAN, right-padded with hexadecimal 'F' characters	cn8
Issuer Public Key File Index	3	Uniquely identifies an Issuer Public Key file	b
Leftmost Digits of the Subject Public Key	$N_I - 36$	$N_I - 36$ most significant bytes of the Issuer Public Key Modulus	b
Subject Public Key Remainder	36	36 least significant bytes of the Issuer Public Key Modulus	b
Subject Public Key Exponent	1 or 3	Issuer Public Key Exponent – value 3 or $2^{16}+1$	b

For an Issuer Public Key, the 20-byte hash code is referred to as the 'Issuer Public Key Checksum'. The EMV2000 specifications do not define this data element.

Transmission of Hash Codes

The hash code transfer file of the Payment System Public Key can be generated by the Member Request Tool to maintain compatibility with the previous public key distribution mechanism. The combination of the self-signed Payment System Public Key and its hash code can be used to ensure dual control when introducing Payment System Public Keys into a member's security infrastructure.

The hash code of the Issuer Public Key transfer format is not used as such, but the 20-byte hash code are displayed for validation during the generation of the Issuer Public Key certification request by the Member Request Tool.

Table 7.3 lists the set of four data fields that constitutes the transfer format for electronically transferred hash codes for MasterCard Public Keys.

Table 7.3—Transfer Format: MasterCard Public Key Hash Code and Associated Data

Field Name	Length	Description	Format
ID of Certificate Subject	5	The MasterCard RID	b
MasterCard Public Key Index	1	Uniquely identifies a MasterCard Public Key	b
MasterCard Public Key Algorithm Indicator	1	Indicates the algorithm to be used with the MasterCard Public Key, set to '01' (hex)	b
Certification Authority Public Key Check Sum	20	Hash code for MasterCard Public Key	b

Table 7.4 lists the set of four data fields that constitutes the transfer format for electronically transferred hash codes for Issuer Public Keys.

Table 7.4—Transfer Format: Issuer Public Key Hash Code and Associated Data

Field Name	Length	Description	Format
ID of Certificate Subject	4	Leftmost 3-8 digits from the PAN, right-padded with hexadecimal 'F' characters	cn8
Issuer Public Key File Index	3	Uniquely identifies an Issuer Public Key file	b
Issuer Public Key Algorithm Indicator	1	Indicates the algorithm to be used with the Issuer Public Key, set to '01' (hex)	b
Issuer Public Key Check Sum	20	Hash code for Issuer Public Key	b

8 File Naming Conventions

This chapter describes the file naming conventions for files that contain keys, certificates, and hash codes.

Introduction

This chapter specifies the file naming conventions used for files that contain keys, certificates and hash codes. The types of files involved are:

- Self-signed Payment System Public Keys (in the format given in Table 4-1)
- Self-signed Issuer Public Keys (in the format given in Table 5-1).
- Issuer Public Key Certificates (in the format given in Table 6-1).
- Certification Authority Public Key hash code, i.e. the hash code on the Payment System Public Key (in the format given in Table 7-3).
- Issuer Public Key hash code (in the format given in Table 7-4).

Note

The Member Request Tool allows automated verification of the Certification Authority Public Key hash code. Therefore, the related file type is only needed for compatibility reasons, and is generated by the Member Request Tool in case the member system requires it.

The Member Request Tool protects the integrity of the Issuer Public Keys certificate request. Therefore, Issuer Public Key hash codes are only used in printed form and the related file type is described only for historical reasons.

The same file name prefix must be used for the files containing a self-signed Issuer Public Key, certificate(s) for this public key, and the hash code for this public key; the files will be distinguished by the use of different 3-letter suffixes. Similarly the same file name prefix will be used for the files containing the self-signed Payment System Public Key and the hash code for this public key, and again the two files will be distinguished by the use of different suffixes.

File names suitable for use with Windows operating systems, or other systems that can accept 'long' file names, are defined in this chapter.

Standard File Name Prefixes

File name prefixes are defined for three types of data object:

- Payment System Public Key
- Issuer Public Key
- Issuer Public Key Certificates.

Payment System Public Key Prefixes

The file name prefix used for files containing a MasterCard Public Key (or the corresponding hash code) contains five characters:

- The first three characters are 'MCI' (in capitals) to denote MasterCard International.
- The last two characters are the one-byte MasterCard Public Key Index for this public key, encoded as two hexadecimal digits (most significant digit first).

This file name is suitable for use in all versions of DOS and Windows.

Issuer Public Key Prefixes

The file name prefix used for files containing an Issuer Public Key (or the corresponding hash code) must contain up to 13 characters:

- The first group of up to six characters can be set by the issuer in the Member Request Tool. As a default, the six character prefix is set to "Prefix".
- The first group of characters (above) must be followed by a single 'dash' character, i.e. '–'. This single character separates the Issuer Identifier from the Issuer Public Key Index.
- The last six characters must be the three-byte Issuer Public Key Index for this public key, encoded as six hexadecimal digits (most significant digit first).

This file name is suitable for use in Windows NT, Windows 95, Windows 98 and OS/2.

Issuer Public Key Certificate Prefixes

The file name prefix used for files containing an Issuer Public Key Certificate (or the corresponding hash code) must contain up to 13 characters:

- The first group of up to six characters must uniquely identify the issuer. MasterCard will supply issuers with a six-character code – known as the MasterCard Certification Authority Member ID – for use by an issuer.
- The first group of characters (above) must be followed by a single 'dash' character, i.e. '–'. This single character separates the Issuer Identifier from the Issuer Public Key Index.
- The last six characters must be the three-byte Issuer Public Key Index for this public key, encoded as six hexadecimal digits (most significant digit first).

This file name is suitable for use in Windows NT, Windows 95, Windows 98 and OS/2.

Standard File Extensions

Table 8.1 defines the three-character file name extensions that must be used for the five types of file specified above.

Note **Since more than one Issuer Public Key Certificate may be produced for a single Issuer Public Key, the file extension for such certificates contains an indication of which Payment System Private Key was used to sign the certificate (thus ensuring that each certificate has a unique file name).**

Table 8.1—Standard File Extensions (3-letter Suffix)

Extension	Use
.sep	Self-signed Payment System Public Key
.c??	Issuer Public Key Certificate
.sip	Self-signed Issuer Public Key
.hep	Hash code calculated on a Payment System Public Key
.hip	Hash code calculated on a self-signed Issuer Public Key

The '??' in the '.c??' file extension used for Issuer Public Key Certificate files is the one-byte MasterCard Public Key Index of the key used to sign the certificate, encoded as two hexadecimal digits (most significant digit first).