



DPC: 21/30362324 DC

BSI Group Headquarters

389 Chiswick High Road London W4 4AL

Tel: +44 (0)20 8996 9000

Fax: +44 (0)20 8996 7400

www.bsigroup.com

Date: 20 April 2021

Origin: International

Latest date for receipt of comments: 13 June 2021

Project No. 2017/01973

Responsible committee: IST/12 Financial services

Interested committees:

Title: Draft BS ISO 11568 Financial services — Key management (retail) — Principles, symmetric ciphers and asymmetric cryptosystems, their key management and life cycle.

Please notify the secretary if you are aware of any keywords that might assist in classifying or identifying the standard or if the content of this standard

- i) has any issues related to 3rd party IPR, patent or copyright
- ii) affects other national standard(s)
- iii) requires additional national guidance or information

WARNING: THIS IS A DRAFT AND MUST NOT BE REGARDED OR USED AS A BRITISH STANDARD. THIS DRAFT IS NOT CURRENT BEYOND 13 June 2021

This draft is issued to allow comments from interested parties; all comments will be given consideration prior to publication. No acknowledgement will normally be sent. **See overleaf for information on the submission of comments.**

No copying is allowed, in any form, without prior written permission from BSI except as permitted under the Copyright, Designs and Patent Act 1988 or for circulation within a nominating organization for briefing purposes. Electronic circulation is limited to dissemination by e-mail within such an organization by committee members.

Further copies of this draft may be purchased from BSI Shop <http://shop.bsigroup.com> or from BSI Customer Services, Tel: +44(0) 20 8996 9001 or email cservices@bsigroup.com. British, International and foreign standards are also available from BSI Customer Services.

Information on the co-operating organizations represented on the committees referenced above may be obtained from <https://standardsdevelopment.bsigroup.com/>.

Introduction

This draft standard is based on national and international discussions. Your comments on this draft are invited and will assist in the preparation of the consequent standard.

For international standards, comments will be reviewed by the relevant UK national committee before sending the consensus UK vote and comments to the international committee, which will then decide appropriate action. If the international standard is approved, it is usual for the text to be published as a British Standard.

For national standards, comments will be reviewed by the relevant UK national committee and the resulting standards published as a British Standard.

UK Vote

Please indicate whether you consider the UK should submit a negative (with supporting technical reasons) or positive vote on this draft. Please indicate if you are aware of any reason why this draft standard should not be published as a British Standard.

Submission of Comments

- **Annotated drafts are not acceptable and will be rejected.**
- All comments should be submitted online at <https://standardsdevelopment.bsigroup.com/>. You will need to register in order to comment.

Template for comments and secretariat observations

Date: xx/xx/20xx	Document: ISO/DIS xxxx
------------------	------------------------

1	2	(3)	4	5	(6)	7
MB	Clause No./ Subclause No./Annex (e.g. 3.1)	Paragraph/Figure/ Table/Note	Type of comment	Commend (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
	3.1	Definition 1	ed	Definition is ambiguous and needs clarifying.	Amend to read '...so that the mains connector to which no connection...'	
	6.4	Paragraph 2	te	The use of the UV photometer as an alternative cannot be supported as serious problems have been encountered in its use in the UK.	Delete reference to UV photometer.	

DRAFT INTERNATIONAL STANDARD

ISO/DIS 11568

ISO/TC 68/SC 2

Secretariat: BSI

Voting begins on:
2021-04-20

Voting terminates on:
2021-07-13

Financial services — Key management (retail) — Principles, symmetric ciphers and asymmetric cryptosystems, their key management and life cycle

Banque — Gestion de clés (services aux particuliers)

ICS: 35.240.40

THIS DOCUMENT IS A DRAFT CIRCULATED FOR COMMENT AND APPROVAL. IT IS THEREFORE SUBJECT TO CHANGE AND MAY NOT BE REFERRED TO AS AN INTERNATIONAL STANDARD UNTIL PUBLISHED AS SUCH.

IN ADDITION TO THEIR EVALUATION AS BEING ACCEPTABLE FOR INDUSTRIAL, TECHNOLOGICAL, COMMERCIAL AND USER PURPOSES, DRAFT INTERNATIONAL STANDARDS MAY ON OCCASION HAVE TO BE CONSIDERED IN THE LIGHT OF THEIR POTENTIAL TO BECOME STANDARDS TO WHICH REFERENCE MAY BE MADE IN NATIONAL REGULATIONS.

RECIPIENTS OF THIS DRAFT ARE INVITED TO SUBMIT, WITH THEIR COMMENTS, NOTIFICATION OF ANY RELEVANT PATENT RIGHTS OF WHICH THEY ARE AWARE AND TO PROVIDE SUPPORTING DOCUMENTATION.

This document is circulated as received from the committee secretariat.



Reference number
ISO/DIS 11568:2021(E)

AVA Documents
Document Sharing Hub
© ISO 2021



COPYRIGHT PROTECTED DOCUMENT

© ISO 2021

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Foreword.....	vi
Introduction.....	vii
1 Purpose.....	1
2 Scope.....	1
2.1 General.....	1
2.2 Derived Unique Key Per Transaction	2
2.3 Symmetric Key Distribution using Asymmetric Techniques	2
2.4 Scope Exclusions	2
3 Normative references.....	3
4 Terms and definitions	4
5 Key Management Requirements	13
5.1 General.....	13
5.1.1 Dual Control and Split Knowledge of Secret or Private Keys.....	14
5.1.2 Permissible Key Forms.....	14
5.1.3 Logging	15
5.1.4 Cryptographic Strength	16
5.1.5 Key Locations	17
5.2 Secure Cryptographic Device.....	18
5.2.1 Additional SCD Requirements for Devices Used in SKDAT.....	20
5.3 Additional CA Requirements	20
5.4 Additional RA Requirements	21
5.5 Key Blocks	22
5.5.1 Overview of key blocks.....	22
5.5.2 Key attributes.....	22
5.5.3 Integrity of the key block.....	23
5.5.4 Key and Sensitive Attributes Field	23
5.6 Key Creation.....	24
5.6.1 Symmetric Key Creation.....	24
5.6.2 Asymmetric Key Creation.....	25
5.7 Key Component and Key Share Creation.....	25
5.8 Check Values.....	26
5.8.1 Introduction	26
5.8.2 Symmetric Check Value Calculation	26
5.8.3 Asymmetric Check Value Calculation	27
5.9 Key Distribution	27
5.9.1 Symmetric Key Distribution	27
5.9.2 SKDAT Asymmetric Key Distribution.....	31
5.10 Key Loading	32
5.10.1 General	32
5.10.2 Loading Key Components or Shares	33
5.11 Key Utilization	34
5.11.1 General Key Utilization Requirements	34
5.11.2 Additional Key Utilization Requirements for SKDAT	35
5.12 Key Storage	35
5.12.1 Cleartext Key Component and Share Storage.....	35
5.12.2 Public Key Storage.....	36
5.13 Key Replacement	36

5.14	Key Destruction	37
5.14.1	General	37
5.14.2	Key Destruction from an SCD	38
5.14.3	Destruction of a Key in Cryptogram Form	38
5.14.4	Component and Share Destruction	38
5.15	Key Backup	38
5.16	Key Archiving	39
5.17	Key Compromise	39
6	Transaction Key Management Techniques	41
6.1	General	41
6.2	Method: Master Keys / Transaction Keys	41
6.3	Derived Unique Key Per Transaction	42
6.3.1	DUKPT Key Management	42
6.3.2	Unique Initial Keys	44
6.3.3	AES DUKPT	45
6.3.4	KSN Compatibility Mode	48
6.3.5	Derived Key OIDs	49
6.3.6	Keys and Key Sizes	50
6.3.7	Helper Functions and Definitions	51
6.3.8	Key Derivation Function	52
6.3.9	Derivation Data	53
6.3.10	“Create Derivation Data” (Local Subroutine)	55
6.3.11	Security Considerations	56
6.3.12	Host Security Module Algorithm	57
6.3.13	Transaction-Originating Device Algorithm	61
6.4	Host to Host UKPT	66
	Annex A (Normative) Key and Component Check Values	68
	Annex B (Normative) Split Knowledge During Transport	72
	Annex C (Normative) Trust Models and Key Establishment	74
	Annex D (Informative) Symmetric Key Lifecycle	83
	Annex E (Informative) Asymmetric Key Lifecycle	85
	Annex F (Normative) Approved algorithms	88
F.1	General	88
F.2	Approved algorithms	88
F.2.1	Algorithms approved for public key transport systems are:	88
F.2.2	Algorithms approved for public key agreement systems are:	88
F.2.3	Algorithms approved for digital signatures	88
F.2.4	Approved hash functions are those described in Reference 5	88
	Annex G (Informative) AES DUKPT Pseudocode Notation	89
G.1	Notation Syntax	89
G.1.1	Values	89
G.1.2	Statements	90
G.1.3	Functions	91
	Annex H (Informative) AES DUKPT Test Vectors	92
H.1	General	92

Annex I (Informative) TDEA Derived Unique Key Per Transaction	93
I.1 Use	93
I.1.1 General	93
I.2 Storage Areas.....	93
I.2.1 General	93
I.2.2 PIN Processing	93
I.2.3 Key Management	93
I.3 Processing Algorithms	96
I.3.1 Load Key.....	96
I.3.2 "Request PIN Entry" (External Command)	97
I.3.3 "Cancel PIN Entry" (External Command)	98
I.3.4 "Power On Reset" (External Command)	98
I.3.5 "Set Bit" (Local Subroutine)	99
I.3.6 "Derivation Process" (Local Subroutine)	99
I.3.7 "Triple-DEA Encrypt" (Local Subroutine)	99
I.4 Key Management Technique.....	99
I.5 DUKPT Test Data Examples.....	102
I.5.1 General	102
I.5.2 Variants of the Current Key	104
I.5.3 Initial Sequence.....	105
I.5.4 MSB Rollover Sequence	108
I.5.5 Calculation and Storage of DUKPT Transaction Keys at the Terminal.....	110
I.6 "Security Module" Algorithm For Automatic PIN Entry Device Checking	112
I.7 "Security Module" Algorithm For Automatic PIN Entry Device Checking	113
Annex J (Informative) Roles in Payment Environment.....	115
Annex K (Informative) Roles in Symmetric Key Distribution using Asymmetric Techniques	118
Bibliography	121

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 68, *Financial Services*, Subcommittee SC 2, *Financial Services, Security*.

This edition cancels and replaces the previous editions of ISO 11568 all parts, which has been technically revised, and incorporates revised text from all former parts.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

Retail financial transactions are often transmitted over potentially non-secure channels, which if exploited, may result in fraud. The vast range in value and volume of such transactions exposes participants to severe risks, which may be uninsurable. To protect against these risks, many institutions are employing encryption. The encryption algorithms used are in the public domain. The security and reliability of any process based on these algorithms is directly dependent on the protection afforded to secrets called cryptographic keys.

This standard describes requirements and procedures for the secure management of cryptographic keys used to protect information in a retail financial services environment; e.g. in messages between a card acceptor and an Acquirer. Typical services in the retail financial services domain include point-of-sale (POS) debit and credit authorizations and automated teller machine (ATM) transactions.

This new version of ISO 11568 merges and updates the previous version of ISO 11568 Parts 1, 2, and 4 with significant contributions from ANSI X9.24 Parts 1, 2, and 3.

Notable Changes:

- Combined all parts into single standard
- Fixed Key is no longer included in the permissible methods of transaction key management.
- Required key replacement policy (see Section 5.13)
- Removed cleartext key injection

Banking – Key management (retail)

1 Purpose

This standard governs the management of cryptographic keys that can be used to protect sensitive information in financial services related to retail payments. While it is designed with that environment in mind, it may also be used in unrelated applications.

This standard establishes the minimum requirements and provides guidance for the secure management and application-level interoperability of key operations. For example, such keys could be used for:

- authenticating messages (see References 3, 9, and 3)
- encrypting Personal Identification Numbers (PIN) (see Reference 1)
- encrypting other data
- encrypting or deriving cryptographic keys
- automated symmetric key distribution using asymmetric techniques

2 Scope

2.1 General

This standard specifies the principles for the management of keys used in the retail financial services environment, which includes the interface between:

- a card accepting device and an Acquirer
- an Acquirer and a card Issuer
- an ICC and a card accepting device

The roles within the retail payment environment are described in Annex J.

This standard covers all methods (manual, automated, or any combination thereof) for management of keying material used for financial services such as POS or ATM transactions, messages between terminals and financial institutions, and interchange messages between Acquirers, switches and card Issuers.

Specified herein are the minimum requirements for the management of keying material. Addressed are key storage and all phases of the key management life cycle, including the generation, distribution, utilization, archiving, replacement and destruction of the keying material.

Included are requirements related to the following concepts:

- key separation;
- key substitution prevention;
- key identification;

- key synchronization;
- key integrity;
- key confidentiality;
- key logging and auditing
- key compromise detection.

Requirements associated with hardware used to manage keys have also been included in this standard.

2.2 Derived Unique Key Per Transaction

This standard includes AES DUKPT as a method to derive unique initial DUKPT keys and unique transaction key(s) from a single base key. Keys that can be derived include symmetric encryption/decryption keys, authentication keys, and HMAC (keyed hash message authentication code) keys. AES DUKPT supports the derivation of AES-128, AES-192, AES-256, and TDEA keys from AES-128, AES-192, and AES-256 initial keys.

Implementations using AES DUKPT as described in this standard provide for the generation of unique transactions key(s) from an initial DUKPT key, in such a way that (1) the originating device does not preserve any information that could be used to derive the transaction key after the transaction has been completed, and (2) the Hardware Security Module at the receiving institution can derive the same transaction key(s) with limited information stored in the receiving system. This is an update to the original Derived Unique Key Per Transaction (DUKPT) algorithm method based on TDEA. This update is based on AES and offers a number of other security improvements. Keys can be derived for use with either the AES or TDEA algorithm.

The implementation described in this standard is recommended for new DUKPT implementations. For legacy purposes, the original TDEA DUKPT implementation as described in Annex A of ANSI X9.24 Part 1 – 2009 is included in this document as Annex I.

2.3 Symmetric Key Distribution using Asymmetric Techniques

Compliant implementation of requirements for secure management of symmetric keys requires (among other things) unique key relationships and strict enforcement of dual control and split knowledge processes when handling cleartext keying material. This includes keys deployed to remote devices or established between communicating pairs. Historically, *compliant* implementation of key distribution has been a manually performed, physically on-site process that is difficult to manage, costly, and/or non-existent (i.e., not compliant). An automated rather than manual method of distributing symmetric keys can address these issues and result in improved security of the financial services environment.

The use of public key cryptography and associated asymmetric key algorithms is one solution for automated remote symmetric key distribution, and therefore included in this key management standard. This solution is referred to as Symmetric Key Distribution using Asymmetric Techniques (SKDAT).

The functional roles for SKDAT are described in Annex K.

2.4 Scope Exclusions

This standard does not specifically address internet banking services offered by an issuer to their own customers through that financial institution's website or applications.

This standard does not address using asymmetric keys to encrypt the Personal Identification Number (PIN) or any other data and does not address asymmetric keys managed with asymmetric keys.

This standard is not intended to apply to the management of the keys installed in an ICC during manufacturing, or the initial key established in an ICC during card personalization.

3 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

1. ISO 9564-1, *Financial Services — Personal Identification Number (PIN) management and security — Part 1: Basic principles and requirements for PINs in card-based systems*
2. ISO/IEC 9796-2, *Information technology — Security techniques — Digital signature schemes giving message recovery — Part 2: Integer factorization-based mechanisms*
3. ISO/IEC 9797, *Information technology – Security techniques – Message Authentication Codes (MACs)*
4. ISO/IEC 10116, *Information technology — Security techniques — Modes of operation for an n-bit block cipher*
5. ISO/IEC 10118 (all parts), *Information technology — Security techniques — Hash functions*
6. ISO/IEC 11770 (all parts) *Information technology — Security techniques — Key management*
7. ISO 13491 (all parts), *Financial Services — Secure cryptographic devices (retail)*
8. ISO/IEC 14888-3, *Information technology — Security techniques — Digital signatures with appendix — Part 3: Discrete logarithm-based mechanisms*
9. ISO 16609, *Banking — Requirements for message authentication using symmetric techniques*
10. ISO/IEC 18031, *Information technology -- Security techniques -- Random bit generation*
11. ISO/IEC 18032, *Information technology -- Security techniques -- Prime number generation*
12. ISO/IEC 18033 (all parts), *Information technology — Security techniques — Encryption algorithms*
13. ISO/IEC 19592-2, *Information technology — Security techniques — Secret sharing — Part 2: Fundamental mechanisms*
14. ISO/IEC 19772, *Information technology — Security techniques — Authenticated Encryption*
15. ISO 20038, *Banking and related financial services -- Key wrap using AES*
16. ISO 21188, *Public key infrastructure for financial services – Practices and policy framework*
17. ANS X9.63, *Public Key Cryptography for the Financial Services Industry: Key Agreement and Key Management Using Elliptic Curve-Based Cryptography*
18. ANSI X9.TR-31, *Interoperable Secure Key Exchange Key Block Specification*

19. ANSI X9.TR-34, *Interoperable Method for Distribution of Symmetric Keys using Asymmetric Techniques: Part 1 – Using Factoring-Based Public Key Cryptography Unilateral Key Transport*
20. RFC 3647, *Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework, Internet Request for Comments 3647*, S. Chokhani, W. Ford, R. Sabett, C. Merrill, S. Wu, November 2003
21. ISO/IEC 7812-1, *Identification cards — Identification of issuers — Part 1: Numbering system*

4 Terms and definitions

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- — ISO Online browsing platform: available at <https://www.iso.org/obp>
- — IEC Electropedia: available at <http://www.electropedia.org/>

For the purposes of this document, the following terms and definitions apply.

4.1

Acceptor

see “Card Acceptor”

4.2

Acquirer

the institution (or its agent) that acquires from the card acceptor the financial data relating to the transaction and initiates that data into an interchange system

4.3

Advanced Encryption Standard (AES)

the algorithm specified in Part 3 of Reference 15

4.4

Algorithm

a clearly specified mathematical process for computation; a set of rules which, if followed, will give a prescribed result

4.5

Archived Key

an inactive key that is being saved in a secure manner for a non-operational purpose

4.6

Asymmetric cryptographic algorithm

a cryptographic algorithm that uses two related keys, a public key and a private key; the two keys have the property that, given the public key, it is computationally infeasible to derive the private key

4.7

Asymmetric cryptosystem

cryptosystem consisting of two complementary operations each utilizing one of two distinct but related keys, the public key and the private key, having the property that it is computationally infeasible to determine the private key from the public key

4.8

Authentication Algorithm

the application of a cryptographic process in which output text depends on all preceding input text

4.9**Authentication Element**

a contiguous group of bits or characters which are to be protected by being processed by the authentication algorithm

4.10**Base Derivation Key (BDK)**

key used in a derivation process to generate initial DUKPT keys. The same key is used by the host to derive the current transaction key

4.11**BDK ID**

a 32-bit value that identifies the Base Derivation Key. This was formerly known as the Key Set Identifier (KSI) in the TDEA DUKPT specification

4.12**Card Acceptor**

party accepting the card and presenting transaction data to an Acquirer – See Section J.4

4.13**Card Issuer**

see "Issuer"

4.14**Certificate**

the public key and identity of an entity together with some other information rendered unforgeable by signing the certificate with the private key of the certifying authority, which issued that certificate

4.15**Certificate Authority (CA)**

an entity that creates and assigns certificates

4.16**Certificate Authority System (CA System)**

the infrastructure required to manage, maintain and secure the key pairs and certificates of CA. A CA System will typically include HSMs, firmware, computer equipment, operating systems and software

4.17**Certificate Policy (CP)**

named set of rules that indicates the applicability of a certificate to a particular community and/or class of application with common security requirements

4.18**Certificate Practice Statement (CPS)**

statement of the practices which a certificate authority (4.15) employs in issuing certificates and which defines the equipment, policies and procedures the CA uses to satisfy the requirements specified in the certificate policies that are supported by it

4.19**Certificate Subject**

entity (e.g., SCD) identified in a certificate

4.20

Chain of Custody

the chronological documentation or paper/digital trail showing the custody, controls, transfer, access, and disposition of the controlled item

4.21

Check Value

a non-secret value that is cryptographically related to the key (or component) and is used to verify that the underlying value is as expected. It is possible for different keys or components to have the same check value

4.22

Ciphertext

data in its encrypted form

4.23

Cleartext

data in its original, unencrypted form

4.24

Communicating Pair

two parties (usually institutions) sending and receiving transactions. This is to include alternate processing sites either owned or contracted by either communicating party

4.25

Component Check Value (CCV)

see "Check Value"

4.26

Compromise

in cryptography, the breaching of confidentiality and/or integrity. A violation of the security of a system such that an unauthorized disclosure of sensitive information may have occurred

4.27

Computationally Infeasible

the property that a computation is theoretically achievable but is not feasible in terms of the time or resources required to perform it

4.28

Credentials

identification data for an entity, incorporating at a minimum the entity's distinguished name and public key. NOTE: Additional data can be included

4.29

Cryptographic Algorithm

a set of rules for the transforming of data using a cryptographic key such as:

1. the transformation from cleartext to cipher text and vice versa
2. synchronized generation of keying material
3. computation or verification of a digital signature or Message Authentication Code

4.30

Cryptographic Key

sequence of characters that controls the operation of a cryptographic transformation (e.g., encryption, decryption) [SOURCE: ISO/IEC 11770-1: 2010: 2.12, modified]. See cryptographic algorithm for further details

4.31**Cryptographic Key Synchronization**

the ability for two nodes that cryptographically process a transaction, to determine the identical Transaction Key

4.32**Cryptographic Strength**

the computational cost of the best-known attack against a given cryptographic algorithm and key size. The cryptographic strength is usually a function of key size, but different algorithms may have a different cryptographic key strength even though their key sizes are the same. The strength of a given algorithm and key size may vary depending on the use of the algorithm. For example, uses with few or no plaintext-ciphertext pairs may be more resistant to attack than uses with many pairs

4.33**Cryptoperiod**

the time span during which a specific key is authorized for use, or the time span during which the keys for a given system will remain in use

4.34**Decryption**

a process of transforming cipher text (unreadable) data into cleartext (readable) data

4.35**Derivation**

a cryptographic transformation process that is used to derive one value from another value. It is typically used in the context of this standard to derive one cryptographic key from another key

4.36**Derivation Identifier/ Derivation ID**

a 32-bit value that identifies the specific initial key derived under the Base Derivation Key specified by the BDK ID. This was formerly known as the 19-bit value Device ID (DID) or TRSM ID

4.37**Derivation Key**

a key that is used to cryptographically compute another key using a derivation process

4.38**Derived Unique Key Per Transaction (DUKPT)**

a key management method that uses a unique key for each transaction and prevents the disclosure of any past key used by the transaction-originating device. The unique Transaction Keys are derived from a base derivation key using only non-secret data transmitted as part of each transaction

4.39**Digital Certificate**

asymmetric cryptosystem that provides for the creation and subsequent verification of digital signatures

4.40**Digital Signature**

a cryptographic transformation of data which, when associated with a data unit and accompanied by the corresponding public-key certificate, provides the services of:

- (a) Origin authentication
- (b) Data integrity
- (c) Signer non-repudiation

4.41

Double-Length TDEA Key

TDEA key having a length of 128 bits, consisting of 112 key bits and 16 parity bits, which is typically represented in 32 hexadecimal digits

4.42

Dual Control

a process utilizing two or more separate individuals, operating in concert, to protect sensitive functions or information whereby no single individual is able to use the function or access all the information (such as an entire key) alone. Also, see “Split Knowledge”

4.43

Electronic Direct Key Loading

key loading through a single cable between secure ports of two SCDs or direct contact (e.g., as with an IC card)

4.44

Electronic Network Key Loading

key loading where there is some device or system between the sending and receiving SCDs

4.45

Encryption

a process of transforming cleartext (readable) data into cipher text (unreadable) data for the purpose of security or privacy

4.46

Entropy

a mathematical measure of the amount of unpredictable information

4.47

Exclusive-Or (XOR)

a mathematical operation, symbol “XOR”, defined as:

$0 \text{ XOR } 0 = 0$ $0 \text{ XOR } 1 = 1$ $1 \text{ XOR } 0 = 1$ $1 \text{ XOR } 1 = 0$

Equivalent to binary addition without carry (modulo-2 addition)

4.48

Hardware Security Module (HSM)

an SCD that provides a set of secure cryptographic services, e.g. key generation, cryptogram creation, PIN translation, and certificate signing

4.49

Hash Function

one-way function that maps a set of strings of arbitrary length on to a set of fixed-length strings of bits.

NOTE: A collision-resistant hash function is one with the property that it is computationally infeasible to construct distinct inputs that map to the same output

4.50

Independent communication

process that allows an entity to counter-verify the correctness of a credential and identification documents prior to producing a certificate (e.g., call-back, visual identification, etc.)

4.51

Initial DUKPT Key

a unique key loaded into an SCD that has been derived from a DUKPT BDK

4.52**Initial Key ID**

a 64-bit value that identifies the specific initial key derived under the Base Derivation Key. It is a concatenation of the BDK ID and the Derivation ID

4.53**Interchange**

mutual acceptance and exchange of messages between institutions for card payment transactions

4.54**Issuer**

the institution holding the account identified by the primary account number (PAN) – See Section Annex J for additional information

4.55**ISO**

International Organization for Standardization

4.56**KDH Binding**

maintaining an exclusive relationship between a KRD and a single authorized KDH (for a given purpose)

4.57**Key**

see “Cryptographic Key”

4.58**Key Agreement**

process of establishing a shared secret key between entities in such a way that neither of them can predetermine the value of that key

4.59**Key Check Value (KCV)**

see “Check Value”

4.60**Key Component**

one of at least two parameters having the format of a cryptographic key that is combined using XOR with one or more like parameters to form a cryptographic key. A component is equal in length to the resulting key

4.61**Key Distribution Host (KDH)**

the host distributing keys in the SKDAT key distribution methodology

4.62**Key Encrypting Key (KEK)**

a key used exclusively to encrypt and decrypt other keys

4.63**Key Injection Facility (KIF)**

a physically secure room in which cryptographic keys are loaded into SCDs by creating the key within one SCD and transferring the cleartext value to another SCD. Maintaining the secrecy of the key being transferred depends upon the physical security of the room and procedural controls for the process

4.64

Key Receiving Device (KRD)

the device receiving keys in the SKDAT key distribution methodology

4.65

Key Separation

a process for ensuring that a key is used for only its intended purpose

4.66

Key Serial Number (KSN)

the concatenation of the DUKPT Initial Key ID and the transaction counter

4.67

Key Set

a group of keys that are all determined by a common cryptographic procedure and differentiated by non-secret input to this procedure such that knowledge of one key does not disclose any other key in the group

4.68

Key Set Identifier (KSI)

a non-secret value that uniquely identifies a key set

4.69

Key Shares

the result of dividing a key into some number (n) of pieces (shares), such that a designated minimum number (m) of pieces are required to reconstitute the key. Each key share is constructed in such a manner that access to fewer than m shares does not disclose any information about the key. Often referred to as an "m of n scheme". In all cases, m is greater than 1, and less than or equal to n

4.70

Key Validation

the procedure whereby the receiver of a public key checks that the key conforms to the arithmetic requirements for such a key in order to thwart certain types of attacks. Also referred to as public key validation

4.71

Keying Material

the data (e.g., keys, attributes, and initialization vectors) that comprise a complete cryptographic key and its relevant metadata

4.72

Master Key

in a hierarchy of Key Encrypting Keys and Transaction Keys, the highest level of Key Encrypting Key is known as a Master Key

4.73

Message Authentication

the act of determining that a message has not been changed since leaving its point of origin. The identity of the originator is implicitly verified

4.74

Message Authentication Code (MAC)

a cryptographic value used to confirm that the message came from the stated sender and has not been changed in transit. see References 3, 9, and 14 for further information

4.75**Node**

any point in a network that does some form of processing of data, such as a terminal, Acquirer or switch

4.76**Non-Repudiation**

the service that provides verifiable evidence to substantiate integrity and origin of data, thereby eliminating successful deniability

4.77**Originator**

the person, institution or other entity that is responsible for and authorized to originate a message

4.78**Parity**

the result of a calculation of the number of '1' bits in a string of '0' and '1' bits, that indicates whether the number of '1' bits is odd or even

4.79**Payment Instrument**

any physical payment card, electronic equivalent or other electronic instrument or order for the transmission or payment of money, sold or issued to one or more persons, whether or not the instrument is negotiable. "Payment instrument" does not include any credit card voucher, any letter of credit or any instrument that is redeemable by the issuer in goods or services

4.80**Personal Identification Number (PIN)**

string of numeric digits established as a shared secret between the account owner and the Issuer, for subsequent use to validate authorized card usage

4.81**Primary Account Number (PAN)**

an assigned number, composed of an Issuer identification number, an individual account identification and an accompanying check digit as specified in Reference 21 that identifies the card Issuer and account owner (e.g., cardholder for physical cards)

4.82**Private Key**

the key in an asymmetric key pair which is known only by that entity

4.83**Pseudo-Random Process**

a process that produces a value which is statistically random and essentially unpredictable even though it was generated by a deterministic algorithm

4.84**Random**

a value in a set that has an equal probability of being selected from the total population of possibilities, hence is unpredictable

4.85**Recipient**

the person, institution or other party that is responsible for and authorized to receive a message or package

4.86

Replay

the process of sending a message which contains all or part of a previously sent message, as a method of perpetrating a fraud

4.87

Secure Cryptographic Device (SCD)

a device that provides physically and logically protected cryptographic services and storage (e.g., PIN entry device (PED) or hardware security module (HSM)). The SCD may be integrated into a larger system such as an ATM or POS terminal

4.88

Sender

the person, institution, or other entity transmitting a message or package

4.89

Share

See "Key Shares"

4.90

Split Knowledge

a condition under which two or more parties separately and confidentially have information (e.g., key components) that, individually, convey no knowledge of the resulting combined information (e.g., a cryptographic key)

4.91

Symmetric Key

a cryptographic key that is used in a symmetric cryptographic algorithm (e.g., AES or TDEA). The same symmetric key that is used for encryption is also used for decryption

4.92

Symmetric Key Distribution using Asymmetric Techniques (SKDAT)

key distribution method that uses asymmetric cryptography to protect the keys during transport. See Roles involved in SKDAT in Annex K

4.93

Switch

a node that can route data from a node to other nodes

4.94

Tamper Evident and Authenticable (TEA) Bag

the one-time use packaging that is designed in such a way as to make it infeasible to access the contents without detection and has a pre-printed unique identifier that cannot be changed without detection to allow for authentication. The form of the packaging is not necessarily restricted to a bag

4.95

Tampering

any unauthorized modification that compromises the security properties of a device or security container. Examples include insertion of active or passive tapping mechanisms into a device to determine, record, or modify secret data; or the attempt to penetrate or open a security container, TEA bag, or device

4.96

TEA Bag

see "Tamper Evident and Authenticable Bag"

4.97**Terminal**

a device/system that initiates a transaction

4.98**Transaction**

a series of messages to perform a predefined function

4.99**Transaction Key**

a key used to cryptographically process the transaction. If more than one key is used for different cryptographic functions, each key may be a variant or derivative of the Transaction Key. A Transaction Key is sometimes referred to as a data key, communications key, session key or working key

4.100**Transaction-Originating SCD**

the device that initiates the financial transaction (e.g. a PIN Encryption Device). An intermediate device that processes the transaction is not a transaction-originating SCD

4.101**Triple Data Encryption Algorithm (TDEA)**

algorithm as specified in Reference 12

4.102**Triple-Length TDEA Key**

a TDEA key having a length of 192 bits, consisting of 168 key bits and 24 parity bits, that is typically represented in 48 hexadecimal digits

4.103**Verification**

the process of associating and/or checking a unique characteristic

5 Key Management Requirements

5.1 General

Key generation, storage, loading, transport, use, and destruction are points at which a key management policy is applied to ensure the secrecy of the keys. For an illustration of the symmetric key lifecycle, see Annex D and for the asymmetric key lifecycle, see Annex E.

The impact of a compromise of a given key should be taken into consideration when deciding the policy for management.

It is necessary to employ mechanisms that detect or prevent the unauthorized use of a key, the use of modified or substituted keys, or the replay of keys.

The unauthorized substitution, manipulation, or misuse of stored keys shall be prevented by one or more of the following means:

- physically and procedurally preventing unauthorized access to the key-storage area
- storing a key encrypted and cryptographically bound to its intended use
- ensuring that it is not possible to know the plaintext value of the key

This section includes general requirements that apply to the management of keys.

5.1.1 Dual Control and Split Knowledge of Secret or Private Keys

The management of cleartext components or shares of symmetric (secret) and asymmetric private keys shall ensure dual control and split knowledge of the resulting key. Throughout the lifecycle of a key, an individual shall not have (or have had) access to all cleartext components or a sufficient number of shares to form the key. Each person or group of persons (e.g., Custodian) responsible for each key component or share shall have received adequate training and shall have signed an acknowledgement of their responsibilities to keep secret the key component or share entrusted to them.

The chain of custody for each cleartext component or share shall be documented and such documentation shall be maintained.

Each person or group of persons responsible for injecting cleartext keys from one SCD to another shall have received adequate training and shall have signed an acknowledgement of the responsibilities entrusted to them.

Access to fewer than the number of shares required to reconstruct the plain text symmetric or private key or components required to reconstruct a symmetric key shall give no information about the overall key.

A cleartext key share or component of a symmetric or private key shall be accessible only to the person or group of persons to whom it has been entrusted and only for the minimum duration required.

5.1.2 Permissible Key Forms

5.1.2.1 Permissible Symmetric Key Forms

Symmetric cryptographic keys shall only exist in one or more of the following forms:

1. In a Secure Cryptographic Device (SCD) as specified in Section 5.2
2. If encrypted and outside an SCD:
 - i) as a cryptogram of the key that shall have been created inside an SCD using an AES or TDEA Key Encrypting Key (See Section 5.1.4 for applicable Key Strength, and Section 5.5 for Key Block requirements)
 - ii) as cryptograms of key shares or key components, where the cryptograms shall have been created inside an SCD using an AES or TDEA Key Encrypting Key (See Section 5.1.4 for applicable Key Strength, and Section 5.5 for Key Block requirements)
 - iii) as cryptograms of the key, key shares, or key components, that shall have been created inside an SCD using RSA OAEP or ECIES as defined in Reference 12
3. If not encrypted and outside of an SCD, a key shall exist only in the following forms:
 - i) as two or more full length key components that are managed in accordance with Section 5.1.1, thereby achieving dual control and split knowledge of the key
 - ii) as two or more key shares that are created using a secret sharing method which is perfectly information-theoretically confidential, and defined in Reference 13, and managed in accordance with Section 5.1.1, thereby achieving dual control and split knowledge of the key

A component shall only be associated with a single key, except by chance.

A Key Encrypting Key that protects a large number of keys, such as an Acquirer or Issuer top-level key that is managed as components or shares should be managed so that it requires three people to load the key.

5.1.2.2 Permissible Asymmetric Key Forms

5.1.2.2.1 Asymmetric Private Key Forms

Private Keys shall only exist in one or more of these following forms:

1. Within an SCD as specified in Section 5.2
2. If encrypted and outside an SCD (including transport between SCDs):
 - i) as a cryptogram of the key that shall have been created inside an SCD using an AES or TDEA Key Encrypting Key (See Section 5.1.4 for applicable Key Strength, and Section 5.5 for Key Block requirements)
 - ii) as cryptograms of key shares, where the cryptograms shall have been created inside an SCD using an AES or TDEA Key Encrypting Key (See Section 5.1.4 for applicable Key Strength, and Section 5.5 for Key Block requirements)
 - iii) as cryptograms of the key or key shares that shall have been created inside an SCD using RSA OAEP or ECIES as defined in Reference 13
3. If not encrypted and outside of an SCD, an asymmetric private key shall exist only as two or more key shares that are created using a secret sharing method which is perfectly information-theoretically confidential, and is included in Reference 13, and managed in accordance with Section 5.1.1, thereby achieving dual control and split knowledge of the key

5.1.2.2.2 Asymmetric Public Key Forms

In an asymmetric cryptosystem there is no secrecy requirement for the storage of the public key, but authenticity and integrity of this key shall be ensured before use, preventing substitution or alteration of the public key or associated information without detection.

One or more of the following techniques shall be used to ensure public key integrity:

- sign the public key and associated data using a digital signature system, thereby creating a public key certificate in accordance with Sections 6.3 and 6.4
- create a MAC for the public key and associated data, using an algorithm defined by Reference 9 and a key used only for this purpose
- depend on protections of an SCD (see Reference 7)
- using the key verification techniques as defined in Section 5.8.3 (when the public key is stored in plain text)

5.1.3 Logging

All instances in which cleartext key components or shares are created, received, accessed, transported, loaded, stored, or destroyed, shall be recorded. A policy that defines the length of time such records are

maintained shall exist and be in use. Consideration should be given to the possible need to inspect those records for a finite length of time after the key is removed from service. At a minimum, records shall reflect the date, time, person(s) involved, activity type (e.g., generation, destruction, shipment), reason for activity, SCD Identifier (if applicable) and TEA bag number(s) (if applicable).

Except in the case of intermediate or ephemeral keys that exist only within an SCD, instances where cryptographic keys are created, transported, received, replaced, or destroyed without manual intervention (e.g., dynamic key exchange) should be logged in such a way as to provide evidence that the event occurred. Similarly, instances where cryptograms are created, received, transported, loaded, or stored with manual intervention should be logged in such a way as to provide evidence that the event occurred.

All operations involving Private Keys shall be recorded.

The integrity of the logs shall be preserved.

For CA operations involving Private Keys, logs shall include reference to all hardware used in support of these activities (Hardware reference shall include SCD serial numbers, and a means to distinguish make, and model as a minimum. Other device attributes may also be recorded). The logs may be manual and/or electronic, and the information may be contained in a combination of logs.

Operations involving the Public Key shall be recorded as indicated in the table below:

	Created	Signed	Installed as Trusted Key	KDH Bind	KDH Unbind	Destroyed
CA	✓	✓	✓			✓
KDH	✓	✓	✓	✓	✓	✓
KRD	✓	✓	✓			

Table 1 — Public Key Recording Matrix

Note: Events related to installation/removal of a KDH Public Key in KDH Bind and KDH Unbind are typically logged at the KDH rather than at the KRD, even though the KRD is the device that is receiving/removing the KDH public key.

5.1.4 Cryptographic Strength

Cryptographic strength of keys shall be at least 112 bits. The cryptographic strength of the asymmetric encryption for SKDAT shall be sufficient to render attacks computationally infeasible. A means to track the effective cryptographic strength of a key shall exist.

The strength of a key that is encrypting other keys should be equal or greater than the key being encrypted. A key ever encrypted by a weaker key inherits the cryptographic strength of the weakest key with which it was ever encrypted. Note that this weakening propagates down through a key hierarchy. For example, encrypting an AES 128 key with a triple-length TDEA key reduces the AES key's cryptographic key strength to that of the TDEA key, which is 112 bits. See [1] for more details on cryptographic strength where various algorithms are used.

Implementations of TDEA should employ Unique Key Per Transaction techniques.

5.1.5 Key Locations

A symmetric or private key shall exist in the minimum number of locations as is necessary for operations. The fewer the instances of the key, the stronger the claim of non-repudiation, as there is a lower probability of compromise.

5.1.5.1 Single Purpose Symmetric Key Usage

Each symmetric key shall have an Intended Purpose Designation (IPD). Such IPD shall be used by the SCD to restrict the use of the key to no more than one item from each of the following four classifications:

1. Algorithm with which the key is to be used
 - i. AES
 - ii. TDEA
 - iii. HMAC
2. Cryptographic function
 - i. Encryption/decryption
 - ii. Authenticated Encryption
 - iii. Derivation
 - iv. Message authentication
 - v. Key wrapping
3. Data element or type of information involved
 - i. Accountholder authentication element (e.g., PIN)
 - ii. Keys
 - iii. Other data element not listed above
4. Functional Direction / Mode of Use
 - i. Encrypt/Generate/Wrap
 - ii. Decrypt/Verify/Unwrap
 - iii. Both

The IPD should be used by the SCD to restrict the use of the key for Communicating Pairs to either:

1. Terminal to Host
2. Host to Host

Each key shall be given a single IPD. IPD configurations enforceable by the SCD shall be cryptographically bound to the key.

Keys electronically transmitted from the generating SCD shall have the IPD identified and bound to the key which could be accomplished by setting attributes in a key block.

Keys shall be given the most restrictive IPD consistent with the function the key will be supporting in the implementation (e.g., encryption only, or decryption only as opposed to both encryption and decryption). There are keys that are by nature bi-directional, such as a Master key within an SCD (e.g., Master File Key).

5.1.5.2 Single Purpose Asymmetric Key Usage

Each asymmetric key pair shall have an Intended Purpose Designation (IPD). Such IPD shall be used by the SCD to restrict the use of the key to only one item from each of the three following classifications:

1. Algorithm with which the key is used
2. Cryptographic function (except for self-signed PIN Entry Device key pairs that are self-signed for delivery in a CSR at the manufacturer)
 - i. Encryption

- ii. Decryption
 - iii. Signing
 - iv. Verifying
 - v. Key Agreement
3. Data element or type of information involved, including
 - i. Keys related to confirming SCD legitimacy and/or functionality (i.e., Manufacturer Keys for verification of legitimate parts, code, etc.)
 - ii. Keys related to SKDAT
 - iii. PINs
 - iv. Data

The IPD should be used by the SCD to restrict the use of the key for Communicating Pairs to either:

1. Terminal to Host
2. Host to Host

Each key shall be given a single IPD. IPD configurations enforceable by the SCD shall be cryptographically bound to the key

Keys electronically transmitted from the generating SCD shall have the IPD identified and bound to the key which could be accomplished by setting attributes in a key block.

5.1.5.3 Additional Intended Purpose Designations (AIPD)

In addition to the segmentation required above, Additional Intended Purpose Designations (AIPD) may also be used to provide for segmentation classifications that are enforced by Policy and Procedure, rather than by the SCD.

Examples of AIPD classifications include, but are not limited to:

- Designation as Production or Non-Production.
- Additional data element segmentation, (e.g., separating PAN from other data types, or segmenting an implementation-specific data element)
- BDK segmentation: For entities processing or injecting DUKPT keys or keys for other key-derivation methodologies, this may be done to segment the risk associated with a BDK compromise

5.2 Secure Cryptographic Device

Keys are managed within devices such as PIN Entry Devices, Hardware Security Modules, and Key Loading Devices. These devices shall meet the requirements of a Secure Cryptographic Device (SCD) as described in Reference 7.

Some key management activities necessitate the use of non-SCD devices where their form factors have inherent limitations preventing them from meeting some SCD security requirements, especially tamper-responsiveness. These limitations have associated security risks which must be addressed by restricted usage and additional controls. These devices are Hardware Management Devices (HMD). Examples of HMDs include, but are not limited to:

1. Smart cards used for component or share transport or storage
2. Smart cards containing public/private key pair(s) used to facilitate management of HSMs

3. Devices used to authorize or enable key management functions

HMDs shall only be used in cases where the compromise of a single HMD would not compromise keys or secrets not held within that HMD.

Where there are requirements in this standard that apply to an SCD, such requirements shall also apply to an HMD, unless specifically excluded.

If multiple cleartext components or shares sufficient to form a key are stored or transported within a single SCD other than an HMD, it is equivalent to having a clear key in the SCD, therefore, the SCD shall meet Tamper Responsive requirements in Part 1 of Reference 7. The SCD shall enforce management of the components or shares in such a way as to ensure Split Knowledge of the key in accordance with Section 5.1.1.

When an SCD is used to generate a key or key component, the SCD shall use a random or pseudo-random process such that certain values are not statistically more probable than others from the set of all possible values, in accordance with Reference 10.

When a key is entered as components, the SCD shall combine all the components using an XOR operation. Figure 1 is an example of the XOR process used to combine key components.

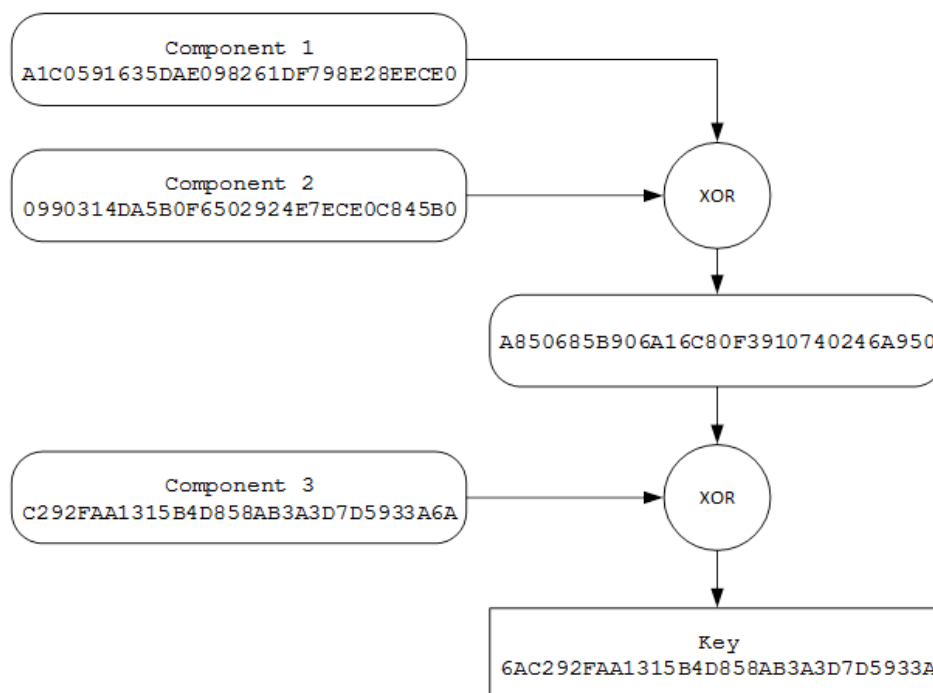


Figure 1 — Example of XOR Function to Combine Key Components

When an SCD is used to produce key shares, the SCD shall use an *m*-of-*n* secret-sharing scheme which is perfectly information-theoretically confidential, and is defined in Reference 13.

The management and control of an SCD shall comply with all parts of Reference 7.

An HMD or an SCD that does not have a Tamper Responsive mechanism enabled, shall be stored in an environment that meets at least the criteria of a Minimally Controlled Environment as identified in Reference 7, with additional controls providing reasonable assurance that any unauthorized access to the

device can be detected. Because the device cannot respond to an attack, the additional controls minimize the probability of unauthorized modifications.

An SCD is decommissioned when it is removed from service for repair, ownership is transferred to another organization, or the SCD is to be destroyed. When an SCD is decommissioned, the financial keys shall be erased in accordance with Part 1 of Reference 7. The keys should be erased as close to the time of removal from service as practical.

When an SCD is lost or stolen, all symmetric or private keys contained in that device should be considered compromised, and the investigation outlined in Section 5.17 shall be initiated.

5.2.1 Additional SCD Requirements for Devices Used in SKDAT

A KRD shall have mechanisms to prevent or detect man-in-the-middle attacks during the execution of SKDAT. A KRD shall have mechanisms that provide secure processing for authentication of the interfacing KDH.

A KDH shall have mechanisms to prevent or detect man-in-the-middle attacks during the execution of SKDAT and the ability to ensure communication with only an intended KRD. A KDH shall have mechanisms that provide secure processing for mutual authentication of the interfacing KRD.

A KRD may interface with more than one KDH for different key types (e.g., one for PIN keys and one for PAN encryption keys offered by a Point to Point Encryption solution provider).

The first time a KRD interfaces with a KDH for a given key type (e.g., delivery of PIN-related keys), such mechanisms may only be sufficient to identify the KDH as having been one of a group of KDHs authorized by the Device Manufacturer or CA for that purpose. Once the KRD is authenticated to the intended KDH for a given key type, mechanisms shall exist to prevent communication with any unintended KDH for that same purpose.

5.3 Additional CA Requirements

Where a CA is used for purposes of verifying the identity of participants in a key exchange protocol, then for certificate issuance the CA shall have processes in place to validate the identity of the certificate requester before issuing a digital certificate for the requester's associated public key.

The CA shall be implemented in accordance with the requirements specified in Reference 16.

The CA shall document a certificate policy (CP) as specified in Reference 20.

The CA shall document a certification practice statement (CPS) in accordance with Reference 20, and the CA shall operate in accordance with its CPS.

In the event of a compromise of the CA System's cryptographic keys, the CA shall have procedures in place to revoke certificates and notify affected parties.

The CA System shall operate in an environment that meets the criteria of a Secure Environment as described in Reference 7 that provides secure processing to ensure the integrity of the public key infrastructure environment.

The CA System Root private key shall only be operated to issue, revoke and suspend subordinate CA certificates. The CA System Root private key shall be operated in a Secure Environment as defined in Reference 7. The CA System Root private key shall be operated off-line, i.e. in an "air-gapped" closed system. Access to the CA System Root private key shall only be under controlled conditions using principles of dual control, observed by an independent witness. Audit / procedure records of each CA

System Root private key access shall be maintained such that all activities performed can be identified from the records alone.

All CA functions, including any registration functions, shall be executed on hardware that is physically separate from the key distribution host hardware.

5.4 Additional RA Requirements

The Registration Authority (RA) is the function that is responsible for identification and authentication of the certificate subject (the entity identified in the certificate), but as it does not perform CA functions, it does not sign or issue certificates. An RA may assist in the certificate application process or revocation process or both. The RA does not need to be a separate body, but can be part of the CA.

The CA shall verify or require that the RA verifies the credentials presented by a certificate subject as evidence of identity or authority to perform a specific role in accordance with the certificate policy and appropriate regulatory and legal requirements. The RA shall verify the certificate subject's possession of the associated private key through the use of a digitally signed certificate request pursuant to PKCS #10 or another cryptographically-equivalent demonstration.

The RA shall also perform validation that the requestor is authorized by doing at least one of the following:

- Confirmation by telephone, confirmatory postal mail, and/or comparable procedure to the certificate subject to confirm that the organization has authorized the certificate application, confirmation of the employment of the representative submitting the certificate application on behalf of the certificate applicant, and confirmation of the authority of the representative to act on behalf of the certificate applicant
- Confirmation by telephone, confirmatory postal mail, and/or comparable procedure to the certificate subject's representative to confirm that the person named as representative has submitted the certificate application
- Confirmation that the requestor is authorized based on previously established trusted communication between the requestor and the RA (e.g., the SCD is directly connected to the Certificate Issuing CA within the manufacturer or repair zone, or sending the public key of a KDH to the CA for signing over an authenticated communication channel)

The RA registration process to validate the certificate subject (SCD) may include, but is not limited to:

- checking the certificate subject's credentials evidencing claimed identity in line with regulatory know-your-customer requirements
- verifying that identifying data provided by the certificate subject is valid
- verifying that the identifying data pertain to the certificate subject
- verifying that the certificate subject is entitled to obtain a public key certificate under the certificate policy
- recording the binding of the certificate subject's identification data to the certificate subject's public key
- checking the certificate request for errors or omissions in accordance with the Certificate Policy

- Direct inspection of the SCD
- Verifying a previously installed certificate
- Verifying that the Serial Number is on a validated list

5.5 Key Blocks

5.5.1 Overview of key blocks

When in storage or transmission, symmetric and private asymmetric keys shall be protected against misuse, substitution and modification through the use of Key Blocks. Public keys can be stored in the same general format of a key block without using encryption.

Note: The use of Key Blocks is required for all symmetric algorithms and key sizes.

The Key Block protects the key and confidential attributes from unauthorized disclosure (using encryption) and protects all data in the Key Block from modification (using integrity protection).

Figure 2 is a high-level illustration of the structure a key block may have but is not intended to be prescriptive.



Figure 2 — Key Block Overview Example

Except in the case of public keys, the key and its confidential attributes shall be encrypted. If the secrecy of the attribute is important to security of the key and key block, it shall be considered a confidential attribute.

The key and its attributes in the key block shall have integrity protection such that they cannot be modified without detection.

The key block shall contain the entire encrypted key (or component or share) and any attributes that are required according to this standard. In addition, either the encrypted or the unencrypted portion of the key block may have additional data that is outside the scope of this standard.

The process to create the Key Block is "key wrapping". Reference 15 is an example of using an AES key to create a key block that meets the requirements in this section 5.5. Reference 18 is an example of using a TDEA key to create a key block that meets the requirements in this section 5.5.

5.5.2 Key attributes

The key attributes define how the key can be used and other pertinent information. These are represented by values known as key tags, which are distinct constants defined in Reference 15 for AES and Reference 18 for TDEA. At a minimum, the attributes shall include the following:

1. One or more attributes as set by the IPD that define the operations for which the key can be used. Examples of attributes that define permitted operations include ones that specify that the key can be used only for PIN block encryption, only for encrypting other keys, or only for computing

message authentication codes (MACs). Other attributes might define sub-categories, for example specifying whether a MAC key can be used to generate a MAC, to verify a MAC, or both

2. One or more attributes that define the cryptographic algorithm and mode for which the key can be used. These attributes are intended to prevent the misuse of a key using a different cryptographic algorithm or mode that could facilitate an attack to determine the value of the key

Security requirements for key attributes are specified in Section 5.5.1.

5.5.3 Integrity of the key block

Before the key in the key block is used, the SCD that will use it shall verify the integrity of the key block with a cryptographic method that uses a key which is at least equal in strength to the key being protected. The key in the key block shall only be used if the integrity check passes.

5.5.4 Key and Sensitive Attributes Field

5.5.4.1 Content of the field

At a minimum, the Key and Sensitive Attributes field of the key block shall contain the value of the key but may also contain other data. For example:

1. Padding that has been used to extend the length of the cleartext key to some desired length before it is encrypted. One purpose of this is to pad a key to the block size of the algorithm used to encrypt it. Another purpose is to hide the true length of the cleartext key, for example by padding so that all keys have the same encrypted length
2. Information indicating the length and location of the key to assist in parsing
3. Any sensitive attributes of that key that need to be kept secret

5.5.4.2 Encryption of the Key and Sensitive Attributes Field

Electronic Code Book mode shall not be used to encrypt this field if the field is longer than one block.

The Key and Sensitive Attributes Field encrypted with AES shall use one of the methods identified in ISO 20038 or Reference 14.

Methods using TDEA keys to encrypt the Key and Sensitive Attributes Field include, but are not limited to:

- method 5 from Reference
- a method defined in ANSI X9.143

For further guidance, see ISO TR 14742 [1].

5.6 Key Creation

5.6.1 Symmetric Key Creation

5.6.1.1 Random Key Generation

Keys shall be generated by using a random or pseudo-random process that ensures that no key generated is more probable than any other key. The strength of the random number generator shall be equal to or greater than the strength of the key it is generating. For further guidance on random number generation, see Reference 10.

Keys shall be generated within an SCD.

5.6.1.2 Key Derivation

Key derivation is a technique by which a (potentially large) number of keys are created ("derived") from a single secret key and non-secret variable data. The original single key is called a "derivation key" or "base key" and each key generated from it is called a "derived key". Key derivation can be used with a variety of key management methodologies and algorithms. For example, DUKPT allows for the derivation of a large number of statistically unique keys from a single key, eliminating the need to store each subsequent derived key. This is useful for an entity that drives a large numbers of devices where each device is required to have unique keys.

Unique Key Per Transaction (UKPT) is an example of a Key Derivation method used between two hosts. UKPT is a key management method to generate a session key based on a base key used by two communication partners exchanging a random value or counter to derive a unique key valid for the transmission of exactly one message or message exchange. It is up to the two organizations involved to determine the information that will be used to derive the unique key(s) for the transaction. See Mechanism 1 in Reference 6 for guidance.

It shall be infeasible to recover the derivation key from any set of derived keys. The procedure may be used iteratively, as a key generated from one initial key may subsequently be used as a derivation key to generate others. The derivation function shall be compliant with Part 2 of Reference 6.

5.6.1.3 Key Calculation (Variants)

Applying a variant is a reversible technique for obtaining a set of keys from a single key, with each resulting key having a different key type. This technique provides key separation while eliminating the need to manage a separate, unrelated key of each required type. Each variant key is calculated from the original key and one constant from a set of non-secret constants using a reversible process. An example of such a process is the modulo-2 addition (i.e., XOR) of the key and a non-secret value. The disclosure of one key calculated in this manner discloses the original key and all other keys calculated from it.

A variant of a key shall be used only for key separation based on usage, and the variant(s) of a key shall only exist in the same SCD that contains the original key.

Where variants are used with Double-Length TDEA keys, and one variant key is used to encrypt data that will also be available in the clear (e.g., PANs), the original key and associated variant keys, shall be replaced regularly, in accordance with Section 5.13.

If a key is compromised, then all variants of the key are also compromised. If there are multiple variants of a key, and any one of those variant keys is compromised, then all variants and the original key are also compromised.

5.6.2 Asymmetric Key Creation

5.6.2.1 Introduction

The asymmetric key pair generation is the process by which a new pair of keys composed of a private key and the related public key is generated for use in a specific asymmetric cryptosystem. The two keys of an asymmetric key pair are mathematically related as defined by the design of the particular asymmetric cryptosystem. The relationship is such that it is computationally infeasible to determine the private key from the public key.

5.6.2.2 Requirements

The following requirements in this section apply regardless of the organization performing the generation activities.

Asymmetric key pairs shall be generated:

1. only within a Secure Cryptographic Device (SCD) as specified in Section 5.2
2. in such a way that guarantees the secrecy of the private key and the integrity of the public key.
3. in accordance with reference(s) applicable to the algorithm(s) used (Reference 6 for RSA and Diffie-Hellman, Reference 17 for Elliptic Curve, and Reference 11 for Prime Number Generator).
4. using an ISO approved method for random number generation (see Reference 10)

If the key pair is generated by a system that will not use it, the following applies to the transfer/transport of the key pair:

1. The private key shall not be available in human-comprehensible form (i.e., it must be encrypted)
2. The private key of the key pair and all related secret seed elements shall be deleted from the sending SCD immediately after the transfer has been confirmed
3. The integrity and confidentiality of the private key shall be ensured throughout the transfer process
4. The public key shall only exist in one of the forms 5.1.2.2.2, and should be delivered within a Certificate if a CA is used

Unique asymmetric key pairs shall be generated for each function (e.g., encryption versus signature) for each entity.

Asymmetric key pairs, when generated, should have a replacement date to establish their life cycle.

5.7 Key Component and Key Share Creation

A symmetric key can exist as two or more full-length components or as a m of n key share scheme with a minimum of two shares, while asymmetric keys do not exist in component form. The processes to create components or shares shall be performed using the principles of dual control and split knowledge. Components can be generated individually and combined to form a key or created from an existing key. The process to manage a key as shares requires the generation of the key and subsequent creation of the shares.

Key components shall be created by using a random or pseudo-random process such that no value is statistically more probable than any other from the set of all possible values. See Reference 12.

Key shares shall be created using a secret sharing method defined in Reference 13, which is identified as perfectly information-theoretically confidential (also known as perfectly information-theoretically secure).

An individual key component shall only be used in the formation of a single key (i.e. the reuse of a component to form more than one key is not permitted).

Key shares and components for a new or existing key shall be created within an SCD.

If component creation or fragmentation into shares produces clear, human readable values, the process shall occur in at least a Controlled Environment as defined in Reference 7.

During the creation process, cleartext components shall only be displayed on or printed by an SCD. Cleartext components produced during creation displayed on the SCD may be recorded on paper by the assigned key custodian.

If components are to be stored or transported, TEA bags shall be used. TEA bags are not required if components are created, loaded, and destroyed in the same activity and location.

5.8 Check Values

5.8.1 Introduction

A Check Value (CV) is used to verify that a value (e.g., a key, key component, or key share) is as expected without disclosing the value itself. At any time following initial generation of the CV, repeating the same function with the same value will produce the same results. If the resulting CV is identical to the initial CV, it is assumed that the value is unchanged.

CV verification can be used to reasonably establish that one or more of the following conditions have been met:

- a value has been correctly entered into a cryptographic device
- a value has been correctly received over a communications channel
- a value has not been altered or substituted

For example, an organization receiving a key, component, or share from another organization may wish to have a method to confirm that the values received have been loaded correctly. CVs generated on keys are called Key Check Values (KCVs). CVs generated on key components are called Component Check Values (CCVs).

Note: Because of the limited size of the CV, in a large set of cleartext values (keys, components, shares), different values can have the same CV (collisions).

If a KCV is used to verify keys and the calculated KCV does not match the expected KCV, the key shall not be used until an investigation is performed to determine the cause of the mismatch.

5.8.2 Symmetric Check Value Calculation

A Check Value should be calculated with a key or component at the time of generation. When a check value is calculated for a key or key component, it shall be calculated by a cryptographic process such that all portions of the key or key component are involved in calculating the check value. It is a significant security risk to calculate check values on portions of the key individually (for example, a separate check

value for the right and left halves of a double length TDEA key). Thus, in the description of this algorithm, the term “Key” always refers to the entire key (for example, all 112 bits of a double-length TDEA key and all 192 bits of a triple-length TDEA key).

When calculating CVs for AES keys, Section A.3 of this standard shall be used. When calculating CVs for TDEA keys, Section A.2 or A.3 of this standard shall be used. When generating a key check value for an HMAC key, it shall be computed according to Section A.4.

When displaying or recording the component check value or key check value computed according to Section A.2, no more than the six hexadecimal digits of the cipher text shall be used.

When displaying or recording the component check value or key check value computed according to Section A.3, no more than the ten hexadecimal digits of the cipher text shall be used.

5.8.3 Asymmetric Check Value Calculation

For asymmetric keys, the reference KCV may be computed as the hash of the public or private key and associated data using an algorithm defined in Reference 5.

For public keys, as long as the KCV is distributed via an integrity-assured channel, the public key can be distributed via a non-secure channel. Prior to installing the public key for use, the user shall validate it by re-computing the KCV and comparing it with the reference KCV.

It shall be infeasible to modify or substitute the reference KCV and public key (and associated data) in such a way that the recomputed KCV of the modified/substituted public key (and associated data) equals the modified/substituted reference KCV. This can be achieved by either of the following:

- by ensuring the integrity of the KCV (see Reference 9)
- by separately storing/distributing the reference KCV in such a way that modification/substitution of the reference KCV cannot be coordinated with modification/substitution of the public key (and associated data) (e.g., dual controls)

5.9 Key Distribution

5.9.1 Symmetric Key Distribution

5.9.1.1 Introduction

The distribution process for an encrypted key shall protect against the use of a key that has been substituted or modified in an unauthorized manner.

For SKDAT, two types of symmetric keys may be involved. The first is the symmetric key (e.g., key encrypting key, PIN encrypting key, DUKPT Initial Key) to be installed into the KRD and corresponding Acquirer host (if applicable), or to be distributed between two hosts. The second is a symmetric key that functions as a key encrypting key for purposes of encrypting a Private Key of a public/private key pair for storage in a database or during transport. The first type is the object of protection, as covered in Section 5.9.1.3, and the second is providing the protection, and is subject to all the requirements of other symmetric keys. A SKDAT implementation shall include a method to bind the symmetric key being distributed to a particular cryptographic algorithm.

For automated SKDAT, examples of acceptable methods of binding the symmetric algorithm to the distributed key include but are not limited to:

- Using cryptographically bound mechanisms e.g. labels, to indicate the appropriate algorithm and usage
- Including the symmetric algorithm name or other unique identifier in the key derivation function of a key agreement scheme

5.9.1.2 Encrypted with Symmetric Key

Keys, components, and shares encrypted with symmetric keys and contained in key blocks in accordance with Section 5.1.2.1 of this standard can be transmitted through unprotected communication channels.

5.9.1.3 Encrypted with Asymmetric Key

Keys, components, and shares encrypted with asymmetric keys as described in Section 5.1.2.1 can be transmitted through unprotected communication channels.

At a high level, there are three phases of SKDAT in typical protocols using a CA. The following description is based on a PIN Entry Device (PED) as the KRD, but the concepts could apply to any KRD.

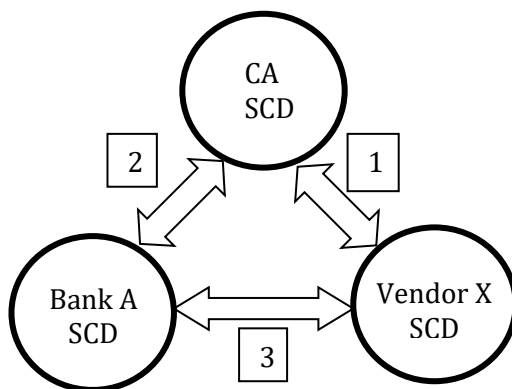


Figure 3 — Automated SKDAT Phases

Phase 1 – KRD Initialization: During the manufacturing phase of the KRD (PED), when the device is initialized (or at a subsequent re-initialization following repair), the KRD vendor will inject data such that the KRD can be reliably distinguished from a counterfeit device. This data typically consists of a certificate, which corresponds to the private key of the KRD. The KRD's key pair may be sent to the KRD encrypted with a transport key, or the KRD may self-generate the key pair and export a Certificate Signing Request (CSR) to the CA, and the CA responds with the certificate.

The KRD vendor will also inject the device with a CA certificate such that the KRD can validate KDH certificates issued by the same CA.

Phase 2 – KDH Initialization: Prior to implementing SKDAT, the same CA certificate that was provided to the KRD is provided to the KDH. Also, similar to what happened in Phase 1 for the KRD, a KDH-specific key pair is generated by or installed (encrypted) into the HSM; the public key of that key pair is signed by the CA; and the signed public key is installed in the KDH (host system, not in the SCD).

Phase 3 – Key Exchange: When the KRD interfaces with a KDH, the KRD and KDH will exchange certificates. Each will use the CA certificate to verify the authenticity of the received certificate.

Phase 1 Requirements:

In Phase 1, the KRD public key that has not yet been signed shall be protected against substitution or modification. Sufficient controls shall exist to ensure that it is infeasible for a counterfeit device to be initialized such that it cannot be reliably distinguished from a genuine device. The CA shall have mechanisms to ensure that the CSR is received from a valid device. One method of achieving this would be to contain the operations where the unsigned public key is presented to the CA for signature. If this method is used, the activity to sign the public key shall occur within a Secure Environment as defined in Reference 7.

Where the root level CA is used to sign a subordinate CA that issues certificates to a KDH or KRD, the authenticity and integrity of the root CA certificate installed with the signed sub-CA public key shall be verified prior to installation in the Secure Environment. The integrity of the CA certificate shall be maintained prior to loading into the KRD. Once the CA certificate is loaded into the KRD, it shall be protected against unauthorized modification, substitution or replay.

The request to generate or sign the KRD certificate shall include sufficient data for the CA to validate the authenticity of the request. The request channel to the CA shall be protected against unauthorized substitution, modification and replay. All requests to the CA to generate or sign certificates shall be logged by the requesting entity, including any requests that may fail. The CA shall log all requests to generate or sign KRD certificates, retaining all element in the CSR, and enough information about the event to facilitate a forensic investigation.

Phase 2 Requirements:

The CA shall validate the KDH CSR in accordance with the Registration Authority requirements as defined in section 5.4. The KDH certificate request to the CA should be sent through an integrity protected channel.

The CA public key shall be distributed as per the appropriate section of reference 16.

The integrity of the CA certificate shall be verified prior to use by the KDH HSM. Once the CA certificate is loaded into the KDH, it shall be protected against unauthorized modification, substitution or replay.

Signing of the KDH public key shall be performed in a Secure environment as defined in Reference 7.

Procedural controls shall ensure that only authorized signed KDH public key(s) are installed for use.

Phase 3 Requirements:

In Phase 3, prior to transport of a symmetric key to the KRD, the KDH shall verify that this is the intended KRD (i. e., the KDH has prior knowledge of the identity of the KRD and does not rely only on the validity of the KRD certificate).

The KRD shall validate the legitimacy of the KDH prior to accepting the symmetric key and installing it for future use.

See Annex C for an overview of asymmetric distribution techniques.

5.9.1.4 Cleartext Key Distribution

Cleartext symmetric keys are distributed as components or shares of the key, or by using an SCD to transport the entire cleartext key. Injection of cleartext symmetric keys is covered in Section 5.10, Key Loading.

Regardless of key distribution method used to transport cleartext components or shares, it shall not be possible for any person to ascertain any of the bits of the final key during the distribution process.

In some cases, it is expedient to have authorized individuals carry (i.e., convey) components or shares of the key from one place to another (e.g., from the point of storage to the point of loading). Requirements for this type of situation are covered in Section 5.9.1.4.1 on Personal Conveyance. The section covering Transporting Components or Shares and Annex B describe the requirements that apply when the transportation of the component(s) or share(s) is not under the physical control of authorized persons at all times (e.g., when mailing).

5.9.1.4.1 Personal Conveyance of Cleartext Components or Shares

When conveying keys, each cleartext component or share not in an SCD shall be in the physical control of only the authorized Custodian and should be in the possession of the authorized Custodian only for the minimum practical time required to transport the component or share, enter it into an SCD (if applicable), and then either securely store or destroy it.

The authorized custodian physically carrying a component or share in printed form shall ensure that the component or share value is not visible to any unauthorized individual during conveyance.

5.9.1.4.2 Transporting Cleartext Components or Shares in an SCD

Cleartext key components or shares can be transported in one or more SCDs. This section covers the transport of a single component or share of one or more keys using an SCD (e.g., a single SCD containing “Custodian 1” components for one or more keys). Transporting multiple components or shares of a given key within a single SCD shall be considered equal to transporting a cleartext key within the SCD and is covered in Section 5.2.

The management of a component or share (including the SCD in which the component or share resides) shall ensure dual control and split knowledge of the resulting key.

For cleartext components or shares transported within an SCD, that SCD shall meet the following requirements in addition to those in Section 5.2:

1. Have logical controls to limit access to authorized individual(s)
2. Have logical controls that include protections against attacks (e.g., repeated failed login attempts)
3. Support authentication controls that render the device or user account unusable and/or remove the keys when the number of invalid authentication attempts exceeds a pre-defined limit

The process to transport the SCD shall comply with Annex B.

5.9.1.4.3 Transporting Cleartext Components or Shares in Printed Form or on Transport Media

Transport Media refers to any device that is used to contain a component or share that does not meet the criteria of an SCD (as defined in Section 5.2), such as an HMD.

The sender and receiver have equal responsibility for the secrecy of the cleartext components or shares that are transported in printed form or on Transport Media. Both parties shall have procedures to ensure the secure handling of the components or shares throughout the process in accordance with the principles of Dual Control and Split Knowledge as described in Section 5.1.1.

The process to transport key components or shares that are in printed form or contained in Transport Media shall be in compliance with Annex B.