NORME INTERNATIONALE

ISO 10202-5

Première édition
1998-07-15

Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré —

Partie 5: Utilisation des algorithmes

Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards —

Part 5: Use of algorithms

(F)

Sommaire

Page

Organisation internationale de normalisation
Case postale 56 • CH-1211 Genève 20 • Suisse
Internet iso@iso.ch

Imprimé en Suisse

Avant-propos

L'ISO (Organisation internationale de normalisation) est une fédération mondiale d'organismes nationaux de normalisation (comités membres de l'ISO). L'élaboration des Normes internationales est en général confiée aux comités techniques de l'ISO. Chaque comité membre intéressé par une étude a le droit de faire partie du comité technique créé à cet effet. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'ISO participent également aux travaux. L'ISO collabore étroitement avec la Commission électrotechnique internationale (CEI) en ce qui concerne la normalisation électrotechnique.

Les projets de Normes internationales adoptés par les comités techniques sont soumis aux comités membres pour vote. Leur publication comme Normes internationales requiert l'approbation de 75 % au moins des comités membres votants.

La Norme internationale ISO 10202-5 a été élaborée par le comité technique ISO/TC 68, *Banque, valeurs mobilières et autres services financiers*, sous-comité SC 6, *Services financiers liés à la clientèle*.

L'ISO 10202 comprend les parties suivantes, présentées sous le titre général *Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré*:

— *Partie 1: Cycle de vie de la carte*

— *Partie 2: Processus de transaction*

— *Partie 3: Relations entre les clés de chiffrement*

— *Partie 4: Modules applicatifs de sécurité*

— *Partie 5: Utilisation des algorithmes*

— *Partie 6: Vérification du porteur de carte*

— *Partie 7: Gestion des clés*

— *Partie 8: Principes généraux et vue d'ensemble*

Les annexes A à H de la présente partie de l'ISO 10202 sont données uniquement à titre d'information.

Introduction

La normalisation de l'architecture de sécurité des systèmes de transactions financières utilisant les cartes à circuit intégré (ICC) est divisée de la manière suivante:

— *Partie 1: Cycle de vie de la carte*

— *Partie 2: Processus de transaction*

— *Partie 3: Relations entre les clés de chiffrement*

— *Partie 4: Modules applicatifs de sécurité*

— *Partie 5: Utilisation des algorithmes*

— *Partie 6: Vérification du porteur de carte*

— *Partie 7: Gestion des clés*

— *Partie 8: Principes généraux et vue d'ensemble*

La présente partie de l'ISO 10202 décrit les processus de chiffrement disponibles pour remplir les fonctions de sécurité définies dans les parties 2, 4 et 6 de l'ISO 10202, et pour lesquelles les algorithmes de chiffrement sont nécessaires.

Elle permet l'exécution de toutes les fonctions de sécurité avec un type d'algorithme symétrique ou asymétrique. Elle ne comprend pas les techniques d'apport de connaissance nul, qui pourront être incluses ultérieurement.

Chaque nœud participant à un processus de chiffrement donné doit avoir la fonction de chiffrement nécessaire.

Les processus de chiffrement nécessaires à l'exécution d'une fonction de sécurité sont spécifiés dans les options. Dans chaque processus, une option séparée est définie pour chaque type d'algorithme. De même, une option particulière est spécifiée pour chaque variante d'un processus de chiffrement nécessitant des étapes de communication supplémentaires.

L'article 5 présente les fonctions de sécurité des processus de chiffrement disponibles pour les effectuer.

L'article 6 spécifie les détails des processus de chiffrement. La présente partie de l'ISO 10202 n'est pas une spécification de mise en application, mais indique les éléments de données nécessaires aux deux nœuds pour garantir que le processus de chiffrement est effectué conformément aux procédures de sécurité exigées.

Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré —

Partie 5:
Utilisation des algorithmes

1 Domaine d'application

La présente partie de l'ISO 10202 s'applique aux échanges de chiffrement dans lesquels au moins un nœud est une ICC ou un SAM. Les échanges entre d'autres nœuds de systèmes ne correspondent pas au domaine d'application de la présente partie de l'ISO 10202.

La fourniture de toute fonction de sécurité est optionnelle et dépend des prescriptions relatives au système. Lorsqu'une fonction spécifique est considérée comme nécessaire, elle doit être exécutée de la manière décrite dans le présent document.

2 Références normatives

Les documents normatifs suivants contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente partie de l'ISO 10202. Pour les références datées, les amendements ultérieurs ou les révisions de ces publications ne s'appliquent pas. Toutefois, les parties prenantes aux accords fondés sur la présente partie de l'ISO 10202 sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des documents normatifs indiqués ci-après. Pour les références non datées, la dernière édition du document normatif en référence s'applique. Les membres de la CEI et de l'ISO possèdent le registre des Normes internationales en vigueur.

ISO 4909, *Cartes bancaires — Zone magnétique — Contenu des données de la piste 3.*

ISO 9564-1, *Banque — Gestion et sécurité du numéro personnel d'identification — Partie 1: Principes et techniques de protection du PIN.*

ISO/CEI 9796, *Technologies de l'information — Techniques de sécurité — Schéma de signature numérique rétablissant le message.*

ISO 10202-1, *Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré — Partie 1: Cycle de vie de la carte.*

ISO 10202-2, *Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré — Partie 2: Processus de transaction.*

ISO 10202-3, *Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré — Partie 3: Relations entre les clés de chiffrement.*

ISO 10202-4, *Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré — Partie 4: Modules applicatifs de sécurité.*

ISO 10202-6, *Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré — Partie 6: Vérification du porteur de carte.*

ISO 10202-7, *Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré — Partie 7: Gestion des clés.*

ISO 10202-8, *Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré — Partie 8: Principes généraux et vue d'ensemble.*

3 Termes et définitions

Pour les besoins de la présente partie de l'ISO 10202, les termes et définitions suivants s'appliquent.

3.1
algorithme asymétrique
algorithme dans lequel les clés de chiffrement et de déchiffrement sont différentes et pour lequel il est impossible, par le calcul, de déduire l'une connaissant l'autre

3.2
certificat
(Voir certificat de clé publique.)

3.3
identificateur de certificat
informations relatives au certificat et permettant la vérification correcte d'un certificat de clé

3.4
texte chiffré
texte en clair chiffré

3.5
résistant à la collision
une fonction est dite résistante à la collision si deux valeurs d'entrée produisent des résultats de sortie différents

3.6
justificatifs d'identité
ensemble des éléments de données attribués à chaque entité et utilisés pour authentifier cette entité

3.7
liaison de chiffrement
deux entités logiques (nœuds) ayant préalablement convenu d'un échange de données et ayant une relation de clé de chiffrement

3.8
nœud de chiffrement
une des deux entités logiques (nœuds) d'une liaison de chiffrement

3.9
déchiffrement
processus de transformation d'un texte chiffré en un texte en clair

3.10
signature numérique
résultat d'une transformation de chiffrement, exécutée par l'initiateur utilisant sa clé secrète avec un algorithme asymétrique, assurant la non-répudiation de la source et l'intégrité des données signées

3.11
nom distinctif
nom identifiant de façon unique une entité au cours d'un processus

3.12
chiffrement
processus de transformation d'un texte en clair en un texte chiffré

3.13
authentification d'entité
confirmation que l'identité d'une entité (nœud) est bien celle déclarée

3.14
identificateur explicite de la clé
voir identificateur de clé

3.15
fonction de tronçonnage
fonction faisant correspondre une chaîne de bits à des chaînes de bits de longueur fixe ayant les deux propriétés suivantes:

— pour une sortie donnée, il est impossible d'obtenir par le calcul une entrée correspondant à cette sortie

— pour une entrée donnée, il est impossible d'obtenir par le calcul une deuxième entrée correspondant à la même sortie

NOTE 1 La bibliographie en la matière contient divers termes qui ont une signification identique ou analogue à fonction de contrôle. Codage comprimé et fonction de condensation en sont des exemples.

NOTE 2 La possibilité de calculer dépend des prescriptions de sécurité et de l'environnement spécifiques de l'utilisateur.

3.16
initiateur
le nœud ou l'entité qui engage un processus

3.17
clé
paramètre utilisé conjointement avec un algorithme de chiffrement pour l'exécution de transformations cryptographiques

3.18
identificateur de clé
information d'une clé permettant au destinataire de déterminer la ou les clés appropriées associées à une transaction

3.19
authentification du message
processus fournissant la preuve cryptographique qu'un message n'a pas été modifié ou détruit sans autorisation

3.20
code d'authentification du message (MAC)
champ de données, dont le contenu peut être utilisé pour vérifier l'intégrité d'un message ou de certains éléments de message

3.21
non-répudiation
service de sécurité fournissant une preuve cryptographique permanente de l'intégrité et de l'origine des données – dans une relation infalsifiable – pouvant être vérifiée à tout moment par une tierce partie

3.22
fonction unidirectionnelle
fonction mathématique qui fait correspondre des valeurs d'entrée à des valeurs de sortie de manière irréversible

3.23
certificat de clé publique (certificat)
ensemble constitué des justificatifs d'identité de l'utilisateur (y compris la clé publique) ainsi que de la signature numérique d'une tierce partie de confiance pour ces justificatifs

3.24
attaque par réflexion
attaque provenant d'un faux répondeur et par laquelle l'initiateur est interrogé dans une session séparée avec la même valeur aléatoire que celle qu'il a transmise dans une session concurrente pour authentifier ce répondeur

3.25
répondeur
le nœud ou l'entité qui répond à l'initiateur d'un processus

3.26
algorithme symétrique
méthode cryptographique utilisant la même clé secrète pour le chiffrement et le déchiffrement

3.27
degré d'actualité des données
méthode permettant d'empêcher un message valide d'être répété ultérieurement en utilisant, par exemple, des informations comme une question demandant une réponse correcte et d'actualité

3.28
jeton
ensemble d'éléments de données formé pour chaque échange de données et émis d'une entité vers une autre.

3.29
code de certification de transaction
résultat du processus de certification de transaction produisant une signature électronique, pouvant être un MAC (basé sur un algorithme symétrique) ou une signature numérique (basée sur un algorithme asymétrique)

3.30
certification de transaction
processus fournissant une preuve cryptographique de l'origine et de l'intégrité des données de transaction pouvant être vérifiée par une tierce partie

3.31
tierce partie de confiance
entité généralement accessible, connue et ayant la confiance des entités impliquées dans une communication

3.32
texte en clair
données intelligibles ayant une signification et pouvant être lues ou traitées sans l'application d'une transformation

NOTE Le paragraphe 3.32 n•existe pas dans la version anglaise de la présente partie de l'ISO 10202.

4 Notations

La présente partie de l'ISO 10202 utilise les notations suivantes.

4.1 Valeurs et entités

Les valeurs et entités sont imprimées en italique:

A nom distinctif de l'entité *A*

***Cert*$_X$** certificat de l'entité *X*

CID_X identificateur de certificat de l'entité *X* (voir annexe B)

$Cred_X$ justificatif d'identité de l'entité *X* (voir annexe A)

k_X une clé (K_X, S_X, P_X) associée à une entité *X*

K_X une clé secrète associée à une entité *X* devant être utilisée avec un algorithme symétrique

KID_{kX} identificateur de clé explicite pour une clé *k* d'une entité *X* (voir annexe B)

KID_{pX} identificateur de clé explicite pour une paire de clés S_X/P_X de l'entité *X* (voir annexe B)

PBF0 bloc PIN de format 0 conforme à l'ISO 9564

PBF1 bloc PIN de format 1 conforme à l'ISO 9564

R_X valeur aléatoire émise par l'entité *X*

S_X/P_X clé publique/secrète associée à une entité *X* devant être utilisée avec un algorithme asymétrique

TP nom distinctif de la tierce partie de confiance

T_{val} période de validité des justificatifs d'identité

T_X timbre à date émis par une entité *X*

Z//Z* concaténation des chaînes binaires *Z* et *Z**

<>I<> séparation des champs

4.2 Processus

Les identificateurs de processus sont en lettres majuscules:

EA authentification d'entité

KE échange de clé

MA authentification du message

ME chiffrement/déchiffrement de message

PV vérification du PIN

TC certification de transaction

4.3 Liste d'option

Les identificateurs d'option sont en lettres minuscules:

a asymétrique

s symétrique

m réciproque

t degré d'actualité des données

4.4 Fonctions

Les identificateurs de fonction sont en lettres minuscules et en italique:

c comparer

d déchiffrer

e chiffrer

g générer une valeur aléatoire

h tronçonner

m authentifier un message

o appliquer une fonction unidirectionnelle

s signer

v vérifier

La notation de la fonction est utilisée de paire avec une indication des valeurs de clés et entités.

c(Y,Z) comparaison de deux chaînes binaires *Y* et *Z*, le résultat étant un code d'état

dK(Z) déchiffrement des données *Z* par un algorithme symétrique utilisant la clé *K*

dS$_X$(Z) déchiffrement des données *Z* par un algorithme asymétrique utilisant la clé secrète S_X

eK(Z) chiffrement des données *Z* par un algorithme symétrique utilisant la clé *K*

eP$_X$(Z) chiffrement des données *Z* par un algorithme asymétrique utilisant la clé publique P_X

R=g() génération d'une valeur aléatoire *R*

h(Z) application d'une fonction unidirectionnelle résistante à la collision qui fait correspondre un élément de données *Z* à une valeur de sortie de longueur fixe utilisant les propriétés publiques (tronçonnage), le résultat du tronçonnage étant *h(Z)*

mK(Z) génération d'un code d'authentification du message (MAC-ing) utilisant un algorithme symétrique comme fonction unidirectionnelle avec la clé secrète *K*, le résultat étant un code d'authentification du message MAC (Message Authentication Code)

vK(MAC) vérification d'un MAC utilisant un algorithme symétrique comme fonction unidirectionnelle avec la clé secrète *K*, le résultat étant un code d'état

oK(Z) application d'une fonction unidirectionnelle qui fait correspondre les données *Z* à une valeur de sortie de longueur fixe utilisant un algorithme avec la clé secrète *K*

sS$_X$(Z) application d'une signature numérique aux données *Z* utilisant la clé secrète S_X (signature), le résultat étant une signature *Sig*

vP$_X$(Sig) processus de vérification de *Sig* utilisant la clé publique P_X, le résultat étant un code d'état

4.5 Signatures numériques

La transmission de *Z* non signé (données à transférer) est obligatoire sauf lorsqu'un schéma de signature numérique avec reprise de message est utilisé (voir ISO/CEI 9796). Dans ce cas, les données signées sont composées de sorte que leurs propriétés structurelles puissent être vérifiées et que *Z* soit récupéré à partir de ces données. Pour cela, *Z* doit être suffisamment court et l'algorithme asymétrique doit être réversible.

Dans la présente partie de l'ISO 10202, la notation $sS_A(Z)$ est utilisée pour la signature avec reprise de données et la signature utilisant une fonction de tronçonnage. Cela signifie que $sS_A(Z)$ peut représenter $sS_A(Z)$ ou $sS_A(h(Z))//Z$ où $h(Z)$ peut indiquer Z.

4.6 Format de message de sécurité

Les messages de sécurité sont des commandes logiques des fonctions de sécurité normalisées. Les informations contenues dans ces messages peuvent être transmises dans le champ associé d'une ICC d'un message 8583 ou interprétées par un SAM ou une application afin de générer les commandes appropriées pour les cartes à circuit intégré (ICC).

Un message de sécurité est identifié de la façon suivante:

- **indicateur de message** identificateur de message logique; concaténation des éléments suivants:

  processus: identificateur de processus (paragraphe 4.2)

  liste d'option: liste des identificateurs d'option (paragraphe 4.3)

  numéro: numéro de message séquentiel

EXEMPLE KEss1

Le message de sécurité contient un ou plusieurs sous-champs de message de sécurité. Les sous-champs obligatoires sont en caractères gras. Le sous-champ < operation > (opération) apparaît au moins une fois dans chaque message.

- **sous-champs de message** < *initiator* > | (initiateur)

  < *respondent* > | (répondeur)

  < *operation* > | (opération)

  < operation >

  ***< initiator >*:** nom distinctif de l'entité source

  ***< respondent >*:** nom distinctif de l'entité de destination

  < operation >= < *Z* > | < *KID* > | < *f(Z)* >

  ***< Z >*:** champ de données facultatif

  ***< KID >*:** identificateur de clé

  ***< f(Z) >*:** résultat de l'application de la fonction *f* aux données *Z* (paragraphe 4.4)

EXEMPLE $A \mid B \mid KID_K \mid KID_{K^*} \mid eK^*(KID_{K^*}) \mid eK(K^*)$

5 Correspondances entre les fonctions de sécurité et les types de processus

Une correspondance entre les fonctions de sécurité telles que définies dans les parties 2, 4 et 6 de l'ISO 10202 et l'ensemble des types de processus est illustrée dans le Tableau 1.

Tableau 1 — Correspondance entre les fonctions de sécurité et les types de processus

FONCTION DE SÉCURITÉ	TYPE DE PROCESSUS		CLÉ
Fabrication de l'ICC			
Fabrication Encartage et initialisation	Chargement de la clé initiale Chargement de la clé initiale		$kMprd_{M-P}$ $kEprd_{E-P}$
Personnalisation de l'ICC			
Personnalisation	Chargement de la clé initiale		$kIctl_{I-C}$
Initialisation de la session de la carte			
Contrôle de compatibilité IC	Aucun processus de chiffrement		
Mise à jour des paramètres CDF			
Activation/désactivation/réactivation/fin CDF Allocation ADF	Échange de clé Échange de clé	KE KE	$kIctl_{I-C}$ $kIctl_{I-C}$
Authentification et vérification propres au CDF			
Vérification du titulaire de la carte Authentification statique de CDF Authentification dynamique de CDF Authentification dynamique de l'hôte émetteur	Vérification du PIN Authentification de l'entité Authentification de l'entité Authentification de l'entité	PV EA EA EA	$kIenc_{I-C}$ [1)] $kIaut_{I}$ $kIaut_{C}$ $kIaut_{I-C}$
Traitement de la transaction CDF			
Autorisation de transaction Confidentialité des données Certification de transaction	Authentification du message Chiffrement de message Certification de transaction	MA ME TC	$kImac_{I-C}$ $kIenc_{I-C}$ [1)] $kIcer_{I-C}$
Sélection ADF			
Sélection ADF	Aucun processus de chiffrement		
Mise à jour des paramètres ADF			
Chargement ADF $KActl_{A-C}$ Activation/désactivation/réactivation/fin ADF	Échange de clé Échange de clé	KE KE	$kIkex_{I-A}$ $kAct_{IA-C}$
Authentification et vérification propres à l'ADF			
Vérification du titulaire de la carte Authentification statique de l'ADF Authentification dynamique de l'ADF Authentification du SAM Authentification du prestataire	Vérification du PIN Authentification de l'entité Authentification de l'entité Authentification de l'entité Authentification de l'entité	PV EA EA EA EA	$kAenc_{A-C}$ [2)] $kAaut_{A}$ $kAaut_{C}$ $kAaut_{S-C}$ $kAaut_{A-C}$
Traitement de la transaction ADF			
Autorisation de transaction Confidentialité des données Certification de transaction	Authentification du message Chiffrement de message Certification de transaction	MA ME TC	$kAmac_{A-C}$ $kAenc_{A-C}$ [2)] $kAcer_{A-C}$
Fin de session de la carte			
Fin de session de la carte	Aucun processus de chiffrement		

Ces clés peuvent être, au choix des émetteurs, générées séparément ou dérivées.

2) Ces clés peuvent être, au choix des fournisseurs, générées séparément ou dérivées.

6 Spécifications de processus

Le présent article spécifie les processus disponibles pour les différentes fonctions de sécurité. Chaque processus peut être utilisé seul ou parallèlement à d'autres processus si nécessaire.

Chaque processus permet l'utilisation de plusieurs options offrant une protection contre différentes menaces telles que l'interception, l'usurpation d'identité, le rejeu, la manipulation ou la répudiation. Il convient de sélectionner les options en fonction de l'analyse du risque provenant des menaces d'un environnement particulier. L'annexe H apporte d'autres indications sur cette partie.

Certains processus incluent des options qui fournissent un degré d'actualité des données lorsqu'il est nécessaire de s'assurer qu'un message est unique et/ou émis à un moment spécifique. L'annexe E explique les différentes façons d'obtenir un degré d'actualité des données.

Le degré d'actualité des données est essentiel pour l'authentification d'entité; la combinaison avec un tel processus garantit automatiquement le degré d'actualité des données.

Si le processus de chiffrement est fondé sur un algorithme symétrique, la clé secrète nécessaire au processus doit avoir été établie au niveau des deux nœuds participants. Si cette clé est partagée entre plus de deux entités appartenant à un groupe, il est cryptographiquement impossible de faire la distinction entre ces entités.

Si le processus de chiffrement est fondé sur un algorithme asymétrique, les informations concernant la clé doivent être générées au niveau de chaque nœud. La clé secrète doit être stockée de manière sûre au niveau du nœud, et les informations publiques correspondantes doivent être certifiées par une tierce partie de confiance fournissant des certificats (voir annexe A).

Tous les éléments de données d'un message doivent être connus au niveau du nœud de destination afin que l'étape suivante du processus puisse être exécutée. Si un élément de données est généré dynamiquement au cours du processus, sa transmission est obligatoire. S'il est déjà connu au niveau du nœud de destination, sa transmission peut être omise.

Les opérations et les paramètres apparaissant en caractères gras sont obligatoires. Tous les éléments obligatoires sont repris dans les flèches des figures. D'autres étapes décrites dans les rôles peuvent ne pas toujours correspondre directement aux étapes décrites dans cette figure.

6.1 Processus 1: échange de clé (KE - Key Exchange)

La transmission électronique sûre d'une clé secrète *K* ou S_B vers ou à partir d'une carte à circuit intégré (ICC) ou d'un SAM doit être effectuée conformément à l'une des options décrites dans le présent paragraphe. On suppose que l'initiateur et le répondeur ont établi une liaison de chiffrement.

La clé secrète qui doit être échangée est toujours émise de *A* vers *B*, sauf lorsque la clé est réciproquement développée par les deux parties impliquées dans la communication.

6.1.1 KE-symétrique-symétrique

Échange d'une clé secrète *K** à utiliser avec un algorithme symétrique au moyen d'un algorithme symétrique (Figure 1).

KEss1 = A | B | KID_K | KID_{K*} | $eK*(KID_{K*})$ | $\mathbf{eK(K*)}$

K clé secrète commune

*K** clé secrète échangée

KID_K identificateur de clé pour la clé *K*

KID_{K*} identificateur de clé pour la clé *K**